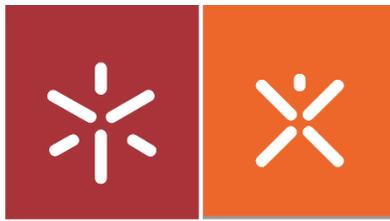


Universidade do Minho
Instituto de Educação

José Carlos Pereira da Silva

**Robótica Educativa, um contributo
para o desenvolvimento do Pensamento
Computacional na escolaridade inicial**



Universidade do Minho
Instituto de Educação

José Carlos Pereira da Silva

**Róbotica Educativa, um contributo
para o desenvolvimento do Pensamento
Computacional na escolaridade inicial**

Dissertação de Mestrado
Mestrado em Ciências da Educação
Área de especialização em Tecnologia Educativa

Trabalho efetuado sob a orientação do
Doutor António José Meneses Osório

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho



Atribuição
CC BY

<https://creativecommons.org/licenses/by-nc/4.0/>

AGRADECIMENTOS

Gostaria de agradecer a várias pessoas, que contribuíram para a realização desta tese de mestrado, e salientar que, sem os seus apoios e incentivos, não conseguiria que esta etapa se tornasse uma realidade. Assim, a saber:

ao Professor Doutor António José Meneses Osório, pela sua orientação e pelo total apoio e disponibilidade. Agradeço o conhecimento transmitido, as suas opiniões e críticas construtivas, que me impulsionaram na elaboração deste trabalho. Importa realçar a colaboração e incentivo para a concretização do trabalho que, por vezes, se mostrou muito difícil terminar pela falta de tempo em articulação com a minha atividade profissional;

agradeço ao Agrupamento, em particular à diretora Rosalina Pinheiro pela pronta aceitação na colaboração com a realização do estudo em algumas turmas do agrupamento;

realço e agradeço a disponibilidade e envolvimento das professoras cooperantes, Carla Barbosa, Sandra Machado e, as restantes colegas, professoras titulares das turmas envolvidas no meu estudo e aos alunos que participaram nas atividades realizadas. Sem eles não seria possível a realização deste trabalho;

aos meus colegas de curso, nomeadamente Raquel Duarte e Rui Barreiro, pelo apoio, companheirismo e incentivo nos momentos mais difíceis;

por último, e não menos importante e tendo plena consciência que sozinho não conseguiria seguir esta caminhada, deixo um agradecimento especial à minha família, pelo seu apoio incondicional, incentivo e paciência demonstrados e total ajuda na superação dos obstáculos.

A eles dedico este trabalho!

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na redação do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas condicentes à sua elaboração.

Mais declaro que conheço e que respeito o Código de Conduta Ética da Universidade do Minho.

RESUMO

O Pensamento Computacional (PC) tem sido apontado por investigadores como um conjunto importante de competências que todos os cidadãos, em particular os alunos, devem desenvolver. Na iniciativa Portugal INCoDe.2030, um dos eixos direcionado à educação, visa precisamente promover a Literacia Digital, a utilização das tecnologias digitais num contexto de inclusão e o desenvolvimento do PC. Portanto, é de esperar que o desenvolvimento do PC tenha uma correlação direta com a aquisição de aptidões que se encontram referidas nos referenciais de competências do séc. XXI, nomeadamente a resolução de problemas e programação. Este trabalho, que apresento, teve como objetivo **caracterizar o processo de desenvolvimento de atividades de Robótica Educativa (RE) com crianças do 1.º ciclo do ensino básico, que promovam dimensões de PC**. Pretendeu-se verificar se, o recurso a conjuntos robóticos, como o LEGO WeDo 2.0, a placa Micro:Bit e o robô Ozobot, através de uma proposta pedagógica, que inclui os conceitos iniciais de programação, é passível de contribuir para promover o desenvolvimento de competências de PC. A opção metodológica seguida foi de natureza qualitativa e interpretativa com um *design* de estudo de caso, de forma a compreender e conhecer as especificidades do desenvolvimento das habilidades do PC em crianças do 3.º e 4.º anos do 1.º ciclo. Para cumprir este estudo, foi realizada uma intervenção pedagógica, com uma forte componente empírica composta por 10 sessões, com aulas orientadas, através de um ensino e aprendizagem centrados no aluno, baseados na abordagem *Use-Modify-Create*, recorrendo à RE para que, de uma forma lúdica, fossem explorados os conceitos básicos das ciências da computação (CC). Através de um modelo quasi-experimental, com avaliação de um pré-teste e de um pós-teste, utilizando o Teste de PC para medir o nível de desenvolvimento do PC dos alunos, bem como uma entrevista coletiva do tipo *focus group* com professores, com intuito de responder ao objetivo geral da investigação. De acordo com a análise e interpretação dos dados, da parte dos professores, existe a perceção de que a utilização da RE refletiu nos alunos maior motivação e interesse, um apelo à criatividade e imaginação, que se adequa a uma aprendizagem baseada na resolução de problemas concretos e promove o pensamento crítico de uma forma ativa. Os resultados do teste de PC, embora ligeiros, apontam para a utilização da RE como uma estratégia pedagógica potenciadora da aprendizagem de conceitos iniciais de programação e do desenvolvimento de competências de PC, embora, como seria de esperar, existam conceitos em que os

alunos revelam mais dificuldade e, portanto, outras estratégias, metodologias e outros estudos são necessários.

Palavras-chave: Programação; 1º Ciclo; Pensamento Computacional; Robótica Educativa

ABSTRACT

Researchers have pointed out that Computational thinking (CT) is an important skill set that all citizens, students in particular, should develop. In the Portugal INCoDe.2030 initiative, one of the educational axes aims precisely to promote digital literacy, the use of digital technologies in a context of inclusion and the development of CT. Therefore, it is to be expected that the development of CT will have a direct correlation with the acquisition of skills that are referred to in 21st century skills frameworks, namely problem solving and programming. The purpose of work we are now presenting was **to characterise the process of developing educational robotics (ER) activities with children in elementary education which promote CT related dimensions**. We intended to verify whether the use of robotic sets, such as LEGO WeDo 2.0, Micro:Bit and the Ozobot robot through a pedagogical proposal, which includes the initial concepts of programming, can contribute to promoting the development of CT skills. The methodological approach followed was qualitative and interpretative in nature with a case study design, in order to understand and get to know the particularities of development CT skills in 3rd and 4th graders. In this study, a pedagogical intervention with a strong empirical component was carried out. This was composed of 10 sessions with guided classes through student-centred teaching and learning, based on the *Use-Modify-Create* approach, using ER to explore the basic concepts of computer science (CS) in a playful way. Through a quasi-experimental model, with evaluation done through a before and after test, using the CT test to measure the level of students' CT development, as well as a teachers focus group in order to respond to the overall objective of the investigation. When analysing and interpreting the data, teachers perceived that the use of ER brought about greater motivation and interest, a call to creativity and motivation, which suits a learning process based on solving concrete problems and promotes critical thinking in an active way. The CT test results, though slight, point to the use of ER as a pedagogical strategy that enhances the learning of initial programming concepts and the development of CT skills, although, as it would be expected, there are concepts that students struggle with. Therefore, other strategies, methodologies and other studies are necessary.

Key words: Programming; Elementary school; Computational thinking; Educational robotics

ÍNDICE

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS	i
AGRADECIMENTOS	ii
DECLARAÇÃO DE INTEGRIDADE.....	iii
RESUMO	iv
ABSTRACT	vi
LISTA DE ABREVIATURAS E SIGLAS.....	ix
ÍNDICE DE FIGURAS	xi
ÍNDICE DE TABELAS.....	xii
ÍNDICE DE GRÁFICOS.....	xiii
1. INTRODUÇÃO	1
1.1. Contextualização do estudo e motivação	2
1.2. Identificação do problema ou problemática e relevância do estudo.....	5
1.3. Questão/Objetivo Geral de investigação	7
1.4. Objetivos do estudo	7
1.5. Estrutura da dissertação	8
2. ENQUADRAMENTO TEÓRICO	9
2.1. Robótica Educativa e Pensamento Computacional, conceitos e princípios.....	10
2.1.1. Robótica Educativa	13
2.1.2. Pensamento Computacional.....	15
2.2. Abordagem <i>Use-Modify-Create</i>	19
2.3. Plataformas de Robótica Educativa.....	22
2.3.1. LEGO WeDo 2.0	26
2.3.2. Placa <i>Micro:Bit</i>	32
2.3.3. Robô Ozobot.....	36
3. METODOLOGIA.....	42
3.1. Opção metodológica	43
3.2. Descrição do estudo.....	44
3.3. Local do estudo.....	46
3.4. Participantes	47
3.5. Materiais pedagógicos	49
3.5.1. Materiais desenvolvidos para o estudo	54
3.5.2. Atividades complementares.....	63

3.6.	Métodos e técnicas de recolha de dados	66
3.6.1.	Teste de Pensamento Computacional	66
3.6.2.	Observação por observação direta.....	71
3.6.3.	Inquéritos por entrevista coletiva do tipo <i>focus group</i>	72
3.7.	Métodos e técnicas de análise de dados	74
3.8.	Calendário de atividades	76
3.9.	Validade, fiabilidade e confiabilidade	77
3.10.	Questões éticas	81
4.	APRESENTAÇÃO E ANÁLISE DE RESULTADOS	84
4.1.	Teste de Pensamento Computacional.....	84
4.1.1.	Procedimentos de recolha	86
4.1.2.	Dificuldades encontradas	88
4.1.3.	Resultados do teste de pensamento computacional	89
4.2.	<i>Focus group</i> professores.....	97
4.2.1.	Apresentação de Resultados.....	98
4.2.2.	Análise dos resultados do <i>focus group</i> de professores.....	99
5.	REFLEXÃO GLOBAL SOBRE OS RESULTADOS	103
6.	CONCLUSÕES	107
6.1.	Considerações finais	108
6.2.	Limitações ao estudo	109
6.3.	Perspetivas futuras	110
7.	REFERENCIAS BIBLIOGRÁFICAS	111
8.	ANEXOS.....	115

LISTA DE ABREVIATURAS E SIGLAS

ACM - *Association for Computing Machinery*

AE – Agrupamento de Escolas

BBC - *British Broadcasting Corporation*

CC – Ciências da Computação

CEB - Ciclo do Ensino Básico.

CNE – Conselho Nacional de Educação

CSTA - *Computer Science Teachers Association*

IMT - *Massachusetts Institute of Technology*

OC - Orientações Curriculares

PC – Pensamento Computacional

PISA - *Programme for International Student Assessment*

ISTE - *International Society for Technology in Education*

RE – Robótica Educativa

RED – Recursos Educativos Digitais

STEAM - *Science, Technology, Engineering Arts and Mathematics*

STEM - *Science, Technology, Engineering and Mathematics*

TIC - Tecnologias da Informação e Comunicação

ÍNDICE DE FIGURAS

Figura 1 - Google Trends – Pesquisa do termo Pensamento Computacional	3
Figura 2 - Conceito conceptual – aluno como mentor e autor	11
Figura 3 – Use-Modify-Create Learning Progression	20
Figura 4 - Artefactos físicos disponíveis comercialmente	24
Figura 5 - Conjunto LEGO WeDo 2.0	27
Figura 6 - Software WeDo 2.0 - Interface em modo Tablet.....	29
Figura 7 - Blocos de Programação LEGO WeDo 2.0	30
Figura 8 - As três fases da metodologia LEGO Education WeDo 2.0	31
Figura 9 – Constituição da placa Micro:Bit V1 e V2.....	33
Figura 10 - Kit básico de eletrónica	34
Figura 11 – Editor Makecode para Micro:Bit.....	35
Figura 12 - Robô Ozobot Evo - Hardware	37
Figura 13 - Atividade com Ozobot, codificação por cores.....	38
Figura 14 - Plataforma de programação - OzoBlockly	39
Figura 15 - Visão global das diversas fases do estudo e instrumentos de validação	44
Figura 16 - Conceitos e práticas fundamentais CSTA	52
Figura 17 - Conjunto de atividades aplicadas durante o programa de intervenção.....	53
Figura 18 - Jogo “Hot Potato” com a placa Micro:Bit	60
Figura 19 - Apresentação sobre o PC	65
Figura 20 - Exemplo I da questão do Teste de PC	69
Figura 21 - Exemplo II da questão do Teste de PC.....	69
Figura 22 - Exemplo III da questão do Teste de PC	70
Figura 23 - Plano de pesquisa para integração da investigação mista.....	74
Figura 24 - Estrutura de desenvolvimento do estudo	76
Figura 25 - Cronograma	77
Figura 26 - Aplicação do Teste piloto.....	79
Figura 27 - Índice do género da amostra.....	85
Figura 28 - Exemplo de uma questão sobre a perceção de autoeficácia.....	85
Figura 29 - Aplicação do Teste de Pensamento Computacional (Pós-teste)	88
Figura 30 - Nuvem de palavras sobre a disciplina OCPR.....	105

ÍNDICE DE TABELAS

Tabela 1 - Horário das sessões de OCPR	45
Tabela 2 - Fases do estudo e objetivos de investigação que se pretendem ver respondidos	46
Tabela 3 - Distribuição da população do Pré-escolar e 1.º Ciclo	47
Tabela 4 - Distribuição de professores	47
Tabela 5 - Distribuição do número de alunos por turma	47
Tabela 6 - Fase de implementação do estudo, resumo das sessões.....	49
Tabela 7 - Estrutura do Framework das Ciências da Computação K-12.....	51
Tabela 8 - Níveis de ensino de acordo com o CSTA	51
Tabela 9 - Conceitos de Ciências da Computação e de Pensamento Computacional	67
Tabela 10 - Distribuição de participantes por género e escolaridade	84
Tabela 11 - Resultados do Teste do Pensamento Computacional em termos percentuais	90
Tabela 12 - Resultados do Teste do Pensamento Computacional	92
Tabela 13 - Médias pré-teste e pós-teste.....	92
Tabela 14 - Resultado dos Testes de Normalidade.....	93
Tabela 15 - Resultado dos Testes de Normalidade do pré-teste e do pós-teste.....	94
Tabela 16 - Ordenações, resultado do teste de Wilcoxon	95
Tabela 17 - Resultado do teste de Wilcoxon - Estatísticas.....	96
Tabela 18 - Quadro Síntese Focus Group.....	98

ÍNDICE DE GRÁFICOS

Gráfico 1 - Teste Wilcoxon – Informações de ordem	96
Gráfico 2 - Teste Wilcoxon – Informações Grupo Pré	96
Gráfico 3 - Teste Wilcoxon – Informações Grupo Pós	96

1. INTRODUÇÃO

“As crianças devem programar um computador (ou outros artefactos), em vez de serem programados por ele”

Procedural Thinking (Papert, 1980, p5)

No ensino atual, tendo em conta a evolução das Tecnologias da Informação e Comunicação (TIC) e o panorama curricular, assiste-se a uma profunda alteração, no seu conteúdo e na sua forma, tornando-se emergente acompanhar estas mudanças, sendo necessário introduzir novas dinâmicas de ensino que incluam tecnologias que permitam rentabilizar o uso de aplicações de *software* complementadas com recursos de *hardware*, em prol de uma mudança de paradigma de aprendizagem.

São observados movimentos e iniciativas na sociedade, nas escolas e instituições educativas a partir da informação disponível em relatórios acerca da introdução do PC nas escolas, tais como: artigos científicos e livros, iniciativas de disseminação e de ensino informal de programação, concursos, redes sociais, internet, publicações nos media nacionais e internacionais, entre outros.

Estes movimentos propõem mudanças, conforme os casos, de novos espaços curriculares com ênfase nas ciências da computação, nomeadamente na inclusão da codificação ou programação, ao longo do percurso escolar de crianças e jovens, um pouco por todo o mundo, com destaque para o contexto europeu onde o nosso país se integra.

Em Portugal, um número significativo de escolas do ensino básico e secundário tem vindo a desenvolver iniciativas e projetos com o objetivo de introduzir o pensamento computacional. Entre as mais recentes, destacam-se as iniciativas por parte do Ministério da Educação, Iniciação à Programação no 1.º CEB, Clubes de Programação e Robótica e neste presente ano letivo 2022/2023, com a inclusão do PC nas Aprendizagens Essenciais no Currículo de Matemática para o Ensino Básico.

A utilização de recursos tecnológicos nas escolas é um fator cada vez mais valorizado em modelos bem sucedidos de ensino, sendo a RE, uma das áreas que cada vez mais, terão influência não só na forma como se ensina e aprende, mas também no próprio funcionamento das empresas do futuro.

A programação e a utilização de artefactos robóticos, permite que os alunos compreendam o que é uma máquina programada e desenvolvam a capacidade de interagir com a realidade, questionem saberes estabelecidos, integrem conhecimentos emergentes e desenvolvam o raciocínio, criatividade, pensamento crítico e abstrato entre outras competências essenciais para o século XXI.

A utilização da robótica educativa no desenvolvimento do pensamento computacional na escolaridade inicial está na génese das preocupações e no propósito deste estudo.

1.1. Contextualização do estudo e motivação

Com o evoluir da tecnologia e a consequente proliferação dos dispositivos, as crianças devem compreender o mundo digital e de que forma podem tirar partido dessa tecnologia em proveito próprio, para que possam contribuir para melhorar o mundo. Compreender o que os computadores ou dispositivos podem ou não fazer será fundamental para lidar com ansiedades. Temos o dever de ensinar as crianças a remodelar o mundo tecnológico à sua volta. De acordo com o ISTE e o CSTA (ISTE & CSTA, 2011), os alunos necessitam de aprender e praticar novas competências para tirar proveito das tecnologias emergentes, no sentido de resolver os desafios do século XXI, não como meros consumidores de tecnologias, mas também como utilizadores que produzam conhecimento e explorem as suas potencialidades para desenvolverem o pensamento crítico, estruturarem e resolverem qualquer tipo de problemas. O PC é um eixo fundamental para o desenvolvimento dessas atividades.

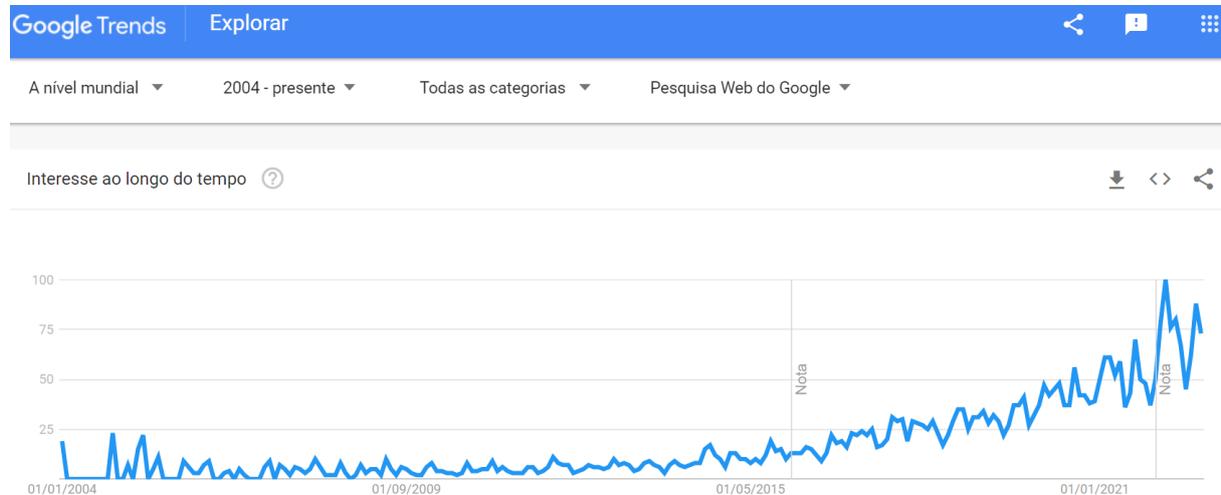
Como o anterior presidente dos Estados Unidos, Barack Obama, disse: “Não compre, apenas, um jogo, crie um. Não se limite a fazer *download* de uma nova aplicação, contribua no seu desenvolvimento. Em vez de jogar no telemóvel, programe-o” (Obama, 2013). Como refere Montessori (2007) no seu trabalho, “a imaginação não se torna grande, até que os seres humanos, se tiverem a coragem e a força, a usem para criar” (p.21). De facto, aprender e brincar não devem ser mutuamente exclusivos, como no âmbito da tecnologia, esta não deve ser apenas para consumir, mas para criar. Montessori acreditava que uma aprendizagem de sucesso estava relacionada com a adaptação de conteúdos às características pessoais de cada criança. Só assim – afirmava – seria possível compreender melhor as crianças. A utilização de objetos tangíveis, pode dar às crianças o poder de serem aventureiras e imaginativas e incentivá-las a tornarem-se criadoras.

Das várias leituras efetuadas, sobressai que o PC descreve o tipo de pensamento que os cientistas das ciências da computação¹ utilizam ao proporcionar estes conhecimentos, recursos e capacidades decorrentes das ciências da computação, para todos os cidadãos, incluindo jovens e crianças. Estas podem ser usadas no quotidiano e são essenciais para os profissionais de qualquer área, Ramos & Espadeiro (2014b). De facto, numa simples pesquisa por/pelo Pensamento Computacional, obtemos

¹ Refere-se ao estudo dos computadores e processos algorítmicos, incluindo os seus princípios, desenho de hardware e software, à sua implementação e impacto na sociedade (CSTA, 2016).

um indicador interessante, através da ferramenta da *Google Trends*, que utiliza o *Big Data* (https://pt.wikipedia.org/wiki/Big_data) para medir as pesquisas, reações e eventos. Podemos observar que o interesse, a nível mundial, pelo PC aumentou vertiginosamente nos últimos anos.

Figura 1 - Google Trends – Pesquisa do termo Pensamento Computacional



Fonte: Imagem retirada pelo investigador, de <https://trends.google.pt/>, em 29/10/2022

O PC tem recebido considerável interesse por parte da comunidade científica e educativa, como a *Computer Science Teachers Association* (CSTA) e a *Association for Computing Machinery* (ACM), com relevância nos conceitos e nas características e as suas implicações para a educação.

Reconhecendo que o PC é abrangente, deve ser abordado de forma motivadora e dinamizadora, através de práticas de sala de aula que envolvam alunos para as atividades relacionadas com estratégias na resolução de problemas, Çoban & Korkmaz (2021), no entanto, por vezes, pouco definido na forma como concretiza as propostas no terreno da escola, principalmente nas diferentes abordagens.

Pretende-se assim recorrer à RE para transmitir os conceitos iniciais de programação, por forma a promover o desenvolvimento de competências do pensamento computacional, em crianças do 1.º CEB.

A utilização da robótica prende-se por experienciar através de um conjunto de recursos robóticos. Portanto ao usar os artefactos robóticos é possível simular a resolução de problemas ou apresentar as soluções para um determinado problema. Por outro lado, a observação de um robô que estamos a programar permite o *feedback* constante para quem está a desenvolver soluções conceptuais no contexto de resolução de problemas.

A motivação que está na origem do estudo desta temática, prende-se por razões pessoais e profissionais, enquanto investigador. Ao longo de todo o percurso académico e profissional, destacam-se três dimensões fundamentais neste processo: o percurso enquanto aluno, como professor e como investigador.

Durante este percurso escolar, foi ganhando forma o desejo de abraçar conceitos mais técnicos e práticos. Aos poucos, foram-se construindo conhecimentos em diferentes áreas técnicas (eletrónica, informática) e um gosto especial pela robótica. No final do secundário, assumiu que queria seguir uma engenharia e enveredar no mundo ligado à técnica.

Enquanto aluno de licenciatura, fruto de vivências em determinadas cadeiras práticas, direcionou-se que o meu futuro passaria por ensinar tecnologias.

Desde sempre, foram utilizados recursos tecnológicos inovadores nas minhas aulas, fruto da evolução tecnológica que acompanha o meu percurso profissional de quase vinte e dois anos.

No ano letivo 2015/2016, fui o responsável pela introdução da disciplina de Oferta Complementar de Programação e Robótica (OCPR) do agrupamento, em Guimarães. Este contacto com alunos do ensino básico representou um passo importante no meu crescimento e desenvolvimento profissional. Por um lado, essa oportunidade de trabalhar com professores e alunos do primeiro ciclo, por outro, aprender a compreensão da própria dinâmica deste nível de ensino e conhece o modo como aprendem as crianças.

Nesta disciplina são trabalhados os conceitos iniciais de programação, promovendo um conjunto alargado de capacidades/competências que importa desenvolver desde muito cedo: o trabalhar em equipa; a estruturação e organização de ideias; a criatividade; o espírito crítico; a resolução de problemas; o pensamento analítico e a atenção aos detalhes.

Por outro lado, a iniciação à robótica complementa as competências desenvolvidas em programação e, além disso, fomenta o gosto pelas ciências, em articulação com as expressões físico-motoras e artísticas, entre outras.

1.2. Identificação do problema ou problemática e relevância do estudo

O PC tem sido apontado, nos últimos anos, como um conjunto importante de competências que todos os cidadãos, e em particular os jovens e crianças, devem desenvolver no século XXI, Wing & Stanzione, (2016) e J. M. Wing, (2017).

Partindo deste *rationale*, em que o pensamento computacional é um conceito fundamental na educação e que envolve a resolução de problemas, conceção de sistemas e compreensão do comportamento humano, baseados nos princípios das ciências da computação, J. M. Wing, (2006). De igual modo, pretende-se através da disciplina de OCPR, existente desde 2016, tirar os melhores contributos decorrentes do caminho já percorrido, refinando e selecionando alguns elementos de maior relevo e propor/ajustar por forma a dar um contributo para o desenvolvimento do PC, nomeadamente através da RE.

Por outro lado, em Portugal as recentes alterações curriculares vieram orientar a integração do PC no currículo dos alunos do 1.º CEB, através das Orientações Curriculares (OC) da componente de Tecnologias de Informação e Comunicação (TIC) e, introduzidas, neste ano letivo (2022/2023), em articulação com as TIC, as Aprendizagens Essenciais da disciplina de Matemática (MEC, 2021), pelo despacho n.º 8209/2021 de 19 de agosto, de acordo com o documento estruturante - Perfil dos Alunos à Saída da Escolaridade Obrigatória.

Nestes documentos, é visível o enfoque no Pensamento Computacional, mais concretamente no envolvimento de alunos em atividades de pensamento computacional e de programação, de forma a promover o desenvolvimento de competências de resolução de problemas, pensamento crítico e criativo, comunicação e colaboração, bem como outras competências relacionadas com o currículo das várias áreas disciplinares.

A relevância do tema não se prende por um alarmismo comum, motivado pelo histerismo tecnológico e as indústrias tecnológicas que as crianças precisam de aprender a programar, porque vamos necessitar de mais engenheiros das ciências da computação. Mas, como educador e como pai questiono, será? E se os alunos, entre os quais, a minha filha não quiserem seguir essa via? Existem outros argumentos que levam a que as crianças devam aprender a programar, para além da moda ou do fator económico. A codificação para crianças é um processo fundamentalmente criativo, assim como cozinhar ou pintar. Com a codificação, uma criança pode manifestar alegria e satisfação ao criar algo, tendo como restrição apenas a sua imaginação. A ideia da Computação Criativa vem enfatizar a importância de não sobrepor

o detalhe técnico e o código ao potencial das atividades de programação na ligação do PC com a criatividade, a imaginação e o interesse dos alunos, *Creative Computing Lab*, (2021).

Ensinar às crianças competências do PC requer persistência, pois envolve conceitos das ciências da computação, como por exemplo a codificação/programação que, por vezes, pode levar a alguma frustração. No entanto, o PC não é programação. Pensar como um engenheiro de *software* significa mais do que escrever código. O PC é uma forma para os seres humanos resolverem problemas, não é tentar fazer com que os seres humanos pensem como computadores, J. M. Wing, (2006). Corroborando as ideias da autora, pensar como um engenheiro de *software* é positivo. Se uma criança aprende algo que não funciona, pode voltar a tentar de diferentes formas até chegar a uma solução. Com a introdução das habilidades do PC, as crianças aprendem a pensar em alternativas quando confrontadas com um determinado problema. Por exemplo, se virar à esquerda não funcionou, então talvez possa seguir em frente ou virar à direita, para se resolver a situação.

Papert defendeu, ao longo da sua vida e da sua obra, a importância do desenvolvimento do PC na educação. Ele acreditava que o uso da tecnologia, nomeadamente o ensino por meio de robôs programáveis e competências do pensamento computacional podiam fornecer às crianças novas possibilidades de aprendizagem, raciocínio e crescimento emocional e cognitivo (Papert, 1980, p.17). Foi também o defensor do princípio que aprendemos fazendo, ou seja, ao programar/o programar o robô para fazer alguma operação, ajuda a criança a pensar como fazer. Por exemplo, ao desenhar um quadrado, a criança tem de saber o que é um quadrado. De facto, Papert acrescentou que o mais importante é aprender a fazer, mas aprendemos melhor se pensarmos sobre o que estamos a fazer, na essência, pois pensamos antes no que queremos executar e damos um passo de cada vez. Devemos pensar antes de agir, ou seja, importa pensar no procedimento, para que possamos, posteriormente, colocar em prática. Desta forma, estamos a elevar o processo de aprendizagem. Esta capacidade de aprender a pensar, o ensinar um computador ou robô a pensar é uma exploração sobre como eles próprios pensam. Daí a justificativa da relevância do tangível no desenvolvimento do PC, mais do que qualquer coisa. O PC é uma ferramenta do pensamento do século XXI, que pode ser aplicada ao longo das nossas vidas e provavelmente com um grande impacto.

Torna-se relevante investigar, descrever e analisar os padrões de participação dos alunos envolvidos com projetos/atividades em que recorrem a RE, no que respeita ao desenvolvimento do PC. Assim, pretende-se estudar a relevância e contributos da Robótica Educativa para compreensão cognitiva de certos aspetos do Pensamento Computacional.

1.3. Questão/Objetivo Geral de investigação

Desde o ano letivo 2015/2016, que o ensino da Programação e Robótica, no 3.º e 4.º anos do 1.º CEB é uma constante, no Agrupamento em que o investigador se insere. Durante estes anos o projeto de Programação e Robótica procurou contribuir para o desenvolvimento de capacidades e competências transversais ao currículo, apesar de nem sempre seguir um trajeto linear.

Sempre que houve essa possibilidade foram usadas metodologias de aprendizagem ativas, procurando estimular aprendizagens, tornando-as mais significativas, possibilitando que os alunos desenvolvessem competências para o século XXI, nomeadamente através da programação e do uso de robôs ou artefactos elétricos/eletrónicos, que fossem capazes de criar histórias, animações, jogos e resolver desafios/problemas do quotidiano.

No entanto, não existem indicadores oficiais de que a disciplina de oferta complementar tenha, na sua globalidade, atingido os seus objetivos e desempenhado um papel relevante no desenvolvimento do pensamento computacional.

Assim, este estudo visa mostrar que através de um programa de intervenção, com atividades com RE é espectável contribuir para o progresso do pensamento computacional em contexto educativo na escolaridade inicial.

Com o propósito de se compreender mais acentuadamente as potencialidades da utilização da robótica educativa na aprendizagem de conceitos em ciências da computação e refletir sobre o contributo nas habilidades do pensamento computacional, foi equacionado o seguinte objetivo geral:

- Caracterizar o processo de desenvolvimento de atividades de Robótica Educativa com crianças do 1.º CEB, que promovam dimensões de Pensamento Computacional.

1.4. Objetivos do estudo

Estando claro o objetivo geral de investigação, identificado no ponto anterior os objetivos específicos irão responder as interrogações fruto da relação entre o campo empírico e teórico e o que queremos alcançar com o estudo.

Para conduzir o desenvolvimento deste estudo, foi seguido um caminho orientado pelos seguintes objetivos específicos:

- Descrever metodologias a implementar;
- Caracterizar materiais pedagógicos que suportem as atividades;
- Identificar os níveis de desenvolvimento do Pensamento Computacional.

Paralelamente importa analisar a implícita transformação dos alunos, nas práticas em que se envolvem e nos novos cenários de aprendizagem criados pelo uso da Robótica Educativa.

1.5. Estrutura da dissertação

Esta dissertação encontra-se dividida em seis capítulos.

No capítulo um fez-se uma breve introdução, onde é dado a conhecer a contextualização e motivação do estudo realizado, identificando o objetivo geral e objetivos específicos do estudo e sua relevância.

O capítulo dois inicia-se com uma breve fundamentação teórica sobre Robótica Educativa e Pensamento Computacional, procurando apresentar resultados de investigações de diferentes autores sobre a temática em análise. Aborda-se a progressão *Use-Modify-Create* (UMC), baseada no princípio e na lógica de que as interações devem ser aprofundadas sucessivamente com suporte apropriado (conceito de *scaffolding*). Ainda nesta secção são apresentadas as plataformas que suportam a programação visual adaptada a crianças do 1.º ciclo, nomeadamente as que estão relacionadas com os principais conjuntos robóticos utilizados no estudo bem como uma breve descrição das principais características desses artefactos (LEGO WeDo 2.0, placa Micro:Bit e Robô Ozobot).

No terceiro capítulo designado por Metodologia, é abordada a opção metodológica, a descrição do estudo, os participantes, materiais pedagógicos utilizados e métodos e técnicas de recolha de dados.

No capítulo quatro, apresentam-se os resultados do estudo realizado no âmbito deste trabalho e apresenta-se uma discussão da sua relevância.

No capítulo seguinte é apresentada uma síntese dos resultados obtidos no trabalho de investigação, invocando autores para uma revisão mais profunda.

Finalmente, no capítulo seis, apresentam-se as conclusões do trabalho com alusão às considerações finais, limitações do estudo e apontam-se caminhos para um possível trabalho futuro.

No final deste estudo, podem ainda ser consultadas as referências bibliográficas, assim como, os anexos que fazem parte da realização do projeto de investigação.

2. ENQUADRAMENTO TEÓRICO

Este capítulo pretende caracterizar os campos da Robótica Educativa (RE) e do Pensamento Computacional (PC) sob diversos aspetos. Faz-se uma breve fundamentação teórica sobre o assunto em estudo, onde são abordadas as principais temáticas do ponto de vista teórico, sendo igualmente apresentados resultados de investigações de diferentes autores. Inicia-se com um conjunto de definições sobre a Robótica Educativa, Pensamento Computacional e a abordagem UMC. Apresentam-se as principais plataformas de Robótica Educativa, com destaque para as que foram usadas neste trabalho: a LEGO WeDo 2.0, a Placa Micro:Bit e o Robô Ozobot.

2.1. Robótica Educativa e Pensamento Computacional, conceitos e princípios

São procuradas evidências do interesse pela Robótica Educativa enquanto instrumento e catalisador do Pensamento Computacional no 1.º Ciclo do Ensino Básico (CEB). O principal objetivo é perceber como a RE pode contribuir, disseminar ou promover os conceitos relacionados ao PC, nomeadamente o desenvolvimento de forma integrada, de práticas como a abstração, a decomposição, o reconhecimento de padrões, a análise ou depuração e os algoritmos.

Pretende-se assim, identificar e abordar as teorias pedagógicas de relevo, no contexto da relevância do tangível no desenvolvimento do PC. Dada a natureza dos ambientes educativos considerados neste estudo, para além da perspetiva construtivista, a ênfase será colocada nas abordagens construcionistas, como paradigmas nucleares na compreensão do campo da Robótica Educativa e do Pensamento Computacional.

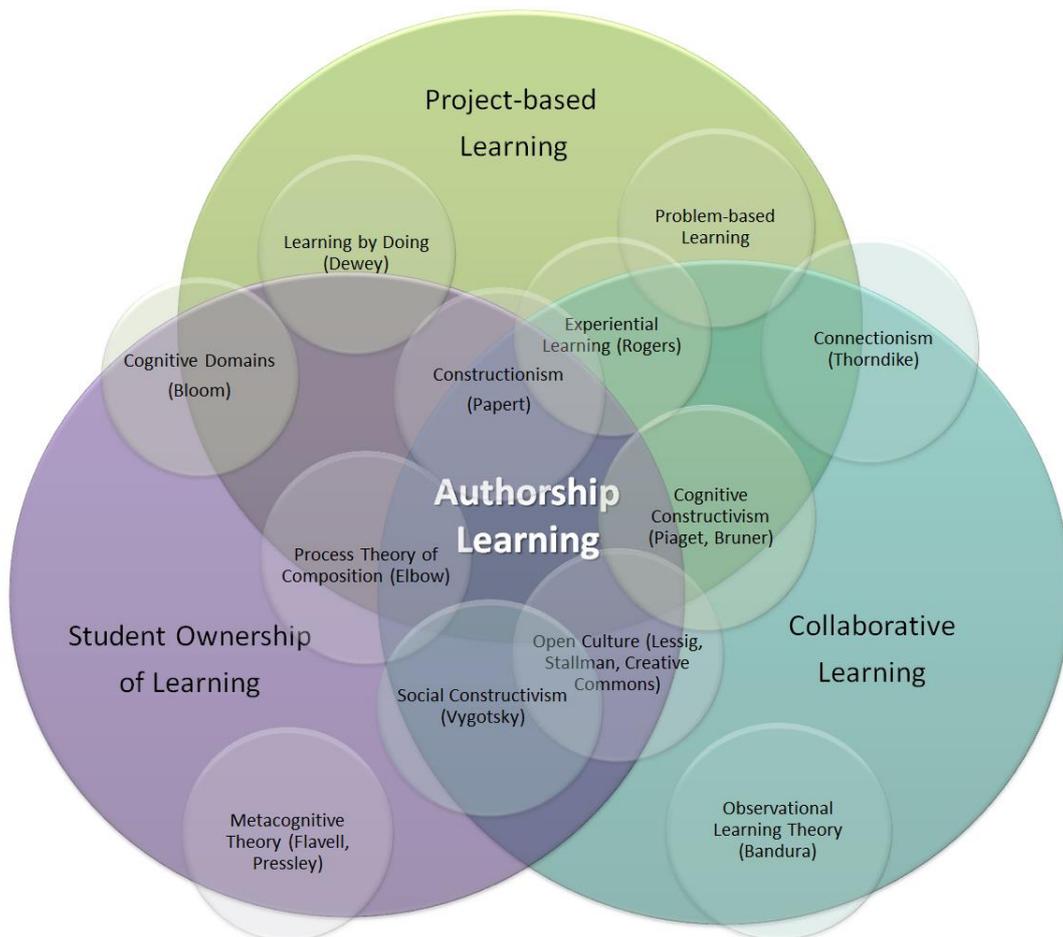
As ideias construtivistas de Piaget identificaram que o conhecimento resultava da interação ocorrida entre as crianças e o meio envolvente, considerando estruturas inerentes e específicas de cada criança. Ou seja, o conhecimento é resultante de uma relação entre as estruturas cognitivas intrínsecas da criança com o meio, Piaget, (1986, p.371). Para este autor, o desenvolvimento intelectual não se constitui apenas por uma adição de conhecimentos e capacidades, mas consiste antes num caminho progressivo, através de diversos estádios qualitativos de desenvolvimento intelectual. Piaget concluiu que a aprendizagem acontece num determinado individuo, partindo de experiências anteriores na tentativa de compreensão do mundo à sua volta. No entanto, observou que as crianças compreendem o que está à sua volta, não na mesma forma que os adultos, mas como agentes ativos que interagem com o mundo e constroem interpretações em constante evolução. Uma parte do construtivismo, assenta na ideia do pensamento por descoberta, que aponta que as crianças aprendem melhor fazendo e explorando.

Na mesma linha de raciocínio, Jérôme Bruner afirma que o objetivo da educação não se remete apenas à aquisição de conhecimento, mas também a melhoria dos processos de pensamento. A descoberta instintiva na resolução de problemas não seria, portanto, um método de ensino e aprendizagem pensado e organizado para conhecer o mundo, e sim para aprender a conhecer o mundo, Bruner (1961).

Mais tarde, estas ideias foram aproveitadas, embora ligeiramente adaptadas e ajustadas, por Seymour Papert, que desenvolveu o conceito intitulado de construcionismo. Baseado no construtivismo de Piaget, o construcionismo propõe a ideia de que todas as crianças aprendem melhor quando são envolvidas na planificação e na própria construção de artefactos ou objetos tangíveis que considerem significativos e

na partilha com os outros. De facto, Papert, ao basear-se no construtivismo, enfatiza a participação dos jovens na construção da sua própria aprendizagem, considerando que a aprendizagem é tendencialmente uma reconstrução e não uma transmissão de conhecimentos. Observou que a utilização de artefactos tangíveis leva a uma aprendizagem mais eficaz, pois o envolvimento da ferramenta induz estímulos cognitivos relevantes, em que os alunos experimentam a construção de um produto com significado, Papert (1986).

Figura 2 - Conceito conceptual – aluno como mentor e autor



Fonte: Imagem retirada pelo investigador, de <https://hybridpedagogy.org/constructionism-reborn/>, em 10/10/2021

De acordo com Trundle & Smith (2017), um professor que promove uma sala de aula baseada no construtivismo, implementa efetivamente uma educação científica prática, direta desenvolvendo nas crianças habilidades cognitivas, sociais, físicas e comunicativas. O ensino deve apostar em ambientes de aprendizagem ativos, com atividades *hands-on*, com recurso a sensores e aplicações de aquisição e análise de dados, *minds-on*, relacionadas com situações complexas, imaginárias ou da vida real e *hearts-on*, com recurso a questões relevantes para os alunos.

As crianças constroem ativamente a compreensão do mundo com base na sua experiência pessoal; ao reconhecer essa participação ativa nos seus próprios processos de aprendizagem, os alunos adquirem alicerces promotores do conhecimento científico, investigativo e crítico.

Um aspecto importante está relacionado com a aprendizagem do próprio aluno enquanto mentor e autor dos seus projetos inovadores, incluindo práticas cognitivas e metacognitivas. Estudos indicam que os alunos aprendem melhor quando constroem os seus próprios significados, Papert (1993, p.139). Essa aprendizagem será mais eficaz se for por um processo de construção física ou digital do mundo real e com aplicabilidade prática. A construção desses artefactos é, muitas vezes, assente na colaboração com outros, por meio de negociação e, por meio de um processo que envolve metacognição sistemática. Nesta perspetiva, Jonan Phillip introduz o conceito de aprendizagem por autor (figura 2), em que os alunos aprendem melhor quando constroem seu próprio significado, Donaldson (2014). O autor refere-se à disponibilidade de uma ampla variedade de ferramentas digitais em que os alunos criam as suas próprias obras, os seus próprios significados e conceitos. Também a partilha dos projetos desenvolvidos permite que o aluno não só possa mostrar como se expressou ao construir o programa, como também pode demonstrar e defender as suas ideias. Isto é fundamental porque os alunos, ao comunicarem estão a ensinar, mostrando as suas ideias e criações, reconhecendo-se como construtores e não como meros consumidores.

No essencial, é salientada a importância para o desenvolvimento das crianças e jovens, de aprender a “desenhar, criar e combinar e não apenas navegar, conversar e interagir” como defende Resnick (2019).

2.1.1. Robótica Educativa

A Robótica Educativa (RE) pode caracterizar-se como sendo "um ambiente de trabalho, onde os alunos têm a oportunidade de montar e programar o seu próprio robô, controlando-o através de um computador com um *software* especializado" (Gonçalves & Freire, 2012, p.1705). Para além disso, com a utilização destes artefactos, como os robôs, pretende-se direccionar o conhecimento para situações reais e do quotidiano dos alunos, Barata & Matos (2019).

Na verdade, sobretudo a partir do trabalho de Wing (2006), os sistemas educativos têm dado cada vez maior importância ao pensamento computacional e à utilização da RE neste processo.

Para além da importância do ensino da Matemática e de outras disciplinas, tem de existir uma interdisciplinaridade entre as diferentes áreas curriculares. Para resolver os problemas da vida real, são necessárias equipas interdisciplinares em vez de especialistas em apenas uma única disciplina, Baker & Galanti (2017). Por outro lado, Kim & Bolger (2017) defendem a criação de um currículo que integre as ciências e a Matemática, devendo-se envolver os professores no desenvolvimento de aulas interdisciplinares adequadas a esta abordagem.

Rodrigues & Felício, (2019, p.111) referem as vantagens destes ambientes de aprendizagem e consideram que: "O desenvolvimento de atividades com robôs torna tudo melhor para os intervenientes na sala de aula: as crianças estão mais motivadas e interessadas; os professores estão mais felizes com o interesse mostrado pelos alunos". Estas potencialidades são referidas por vários outros autores, entre eles, Resnick e Bers quando sugerem que ao construir e programar robôs através da resolução de problema os alunos, mesmo os mais pequenos, desenvolvem competências de pensamento computacional como a resolução de problemas, abstração, decomposição, reconhecimento de padrões e de pensamento lógico e criativo.

Depois do trabalho de Papert, outros estudos foram realizados e apontam na direção de diversos benefícios da utilização da RE em crianças e jovens. Um trabalho de revisão de literatura sistemática foi elaborado pelos investigadores Anwar et al. (2019), referenciando um conjunto de projetos com RE entre 2000 e 2018 em ambientes educativos formais e informais. Um trabalho semelhante da autoria de Neto and Bertagnolli (2021) culminou numa revisão de literatura sistemática mais recente entre 2016 e 2021, sobre a RE, nomeadamente no tipo de abordagens práticas, a formação de docentes e robôs utilizados em práticas educativas.

Chalmers (2018), realizou um estudo em que analisa, como quatro professores de quatro escolas australianas integraram a robótica e programação nas suas atividades letivas, com o objetivo de investigar as percepções dos professores e a influência da incorporação do robô WeDo 2.0 nas suas aulas. No entanto, convém referir que os participantes não receberam qualquer formação/instrução como operar com o conjunto robótico WeDo 2.0. Nesse caso, foi importante procurar compreender os benefícios e as limitações/barreiras da utilização desse conjunto robótico. Os resultados demonstraram que a exploração e o uso dos robôs ajudaram os professores a sentirem-se mais confiantes ao adquirir o conhecimento necessário para promover habilidades do PC aos alunos. No entanto, o conhecimento limitado sobre a forma como ensinar o PC, levaram os professores a enfrentarem desafios técnicos e a sentirem dificuldades na implementação das atividades com robótica, razão pela qual, segundo a autora, existe necessidade de apoio contínuo para que o professor possa integrar robótica e PC nas suas aulas.

As ferramentas utilizadas nos diferentes estudos, diferem de acordo com o público-alvo, orçamento disponível, realidade social e cultural, infraestrutura e local da sua aplicação. Algumas são direcionadas para crianças como o Kibo, (Pinto et al., 2021; Relkin & Bers, 2021), Bee-bot, (Diago et al., 2018), Dash and Dot, (Huang et al., 2018), Cubetto, (Alsina & Acosta, 2022), Ozobot, (Chou, 2018; Körber et al., 2021; Žáček & Smolka, 2019), LEGO WeDo 2.0, (Chalmers, 2018; Demetriadis et al., 2012; Üşengül & Bahçeci, 2020), Placa Micro:Bit, (Rogers & Siever, 2018), e outras são mais avançadas para adolescentes, como o Mbot, Mester (2016) e a Placa Arduino, Cristina & Rocha (2020). O objetivo destas ferramentas é de introduzir conceitos mecânicos, eletrónica e das ciências da computação.

Em unanimidade, os diferentes autores consideram o uso da RE, uma forma eficiente de influenciar as crianças e jovens nas suas habilidades tecnológicas, académicas e sociais, nomeadamente no interesse nas áreas STEAM, no desenvolvimento do pensamento crítico e na resolução de problemas, no trabalho colaborativo e na criação de projetos multidisciplinares.

2.1.2. Pensamento Computacional

O PC é cada vez mais referido em investigações e por educadores desde a sua definição, em 2006, por Wing. A sua importância e relevância é pertinente pela inclusão nos currículos escolares de vários países e pela investigação desenvolvida e publicada sobre o tema.

Tendo em consideração a definição proposta por Wing, outros autores têm vindo a definir PC como um conjunto de aptidões relacionadas com as ciências da computação, tais como a decomposição e resolução de problemas, a abstração, o pensamento lógico/crítico e o reconhecimento de padrões, Brennan & Resnick (2012). Mais recentemente, uma equipa de investigadores do Técnico a Fundação Calouste Gulbenkian e a TreeTree2 (<https://gulbenkian.pt/computacao-e-sociedade>) apresenta que “É a capacidade de formular um problema e expressar soluções executáveis por um computador (seja o computador humano ou máquina). Assim, o Pensamento Computacional é uma competência que permite resolver problemas de uma forma estruturada e lógica.” Segundo Brackmann (2017), “O Pensamento Computacional é uma distinta capacidade criativa, crítica e estratégica humana de saber utilizar os fundamentos da Computação, nas mais diversas áreas do conhecimento, com a finalidade de identificar e resolver problemas, de maneira individual ou colaborativa, através de passos claros, de tal forma que uma pessoa ou uma máquina possam executá-los eficazmente”. Já numa visão mais recente Wing (2014), atualiza e revê a definição para “São os processos de pensamento envolvidos na formulação de um problema e que expressam sua solução ou soluções eficazmente, de tal forma que uma máquina ou uma pessoa possa realizar.”

Das breves pesquisas efetuadas sobre a definição do PC, verifica-se que todas elas elencam pontos em comum. No entanto, no trabalho de Brackmann (2017), faz-se referência a um estudo de dez anos, que não conseguiu uma exata definição para o PC, considerando que ainda existem críticas que sugerem que não sabemos o que o PC significa ou sua forma de medir (p.29). Assim sendo, optou-se por uma definição do PC baseada em Brackmann (2017), por estar direcionada aos objetivos definidos para esta investigação, nomeadamente os relacionados com as ciências da computação. O Pensamento Computacional é uma distinta capacidade humana, criativa, crítica e estratégica, que utiliza os princípios base da Ciência da Computação, nas diversas áreas do conhecimento. Tem como objetivo identificar e resolver problemas, de maneira individual ou colaborativa, através de passos simples que permitem que determinado ser humano ou máquina os execute de forma eficaz.

No entanto, convém esclarecer que o PC é uma capacidade fundamental para todos, não apenas para aqueles que dominam a programação. Segundo Jeannette Wing, para além de aprender a ler, escrever e calcular, deveria ser adicionado o PC na capacidade analítica de cada criança. Por outro lado, apesar das vantagens e potencialidades da robótica educativa é necessário pensar em estratégias e metodologias para a sua integração e utilização em sala de aula.

O PC requer a compreensão de competências básicas das ciências da computação, como a representação de problemas a serem tratados por um computador e a escrita de algoritmos para que o computador possa executar. O contexto e a abordagem mais eficazes para desenvolver o pensamento computacional é aprender ciência da computação, já que eles estão intrinsecamente ligados. O PC está no cerne das práticas das ciência da computação e, de acordo com o documento orientador (*K-12 Computer Science FrameWork*) do CSTA (2016), é delineado pelas seguintes dimensões:

Abstração – é o resultado da redução de um processo ou de um conjunto de informações num conjunto de características importantes para as ciências da computação. Uma abstração pode ser criada para generalizar diferentes situações, retirando propriedades comuns, abstraindo-se de detalhes e implementações específicos e irrelevantes. De facto, esta dimensão pretende extrair a informação essencial de um problema, ou seja, realçar processos relevantes e secundarizar detalhes e especificidades particulares. Esta é uma prática que permite expressar uma solução através da simplificação ou eliminação de detalhes ou aspetos irrelevantes, sendo uma forma eficaz de lidar com a complexidade. Como exemplo, usamos em diversas situações representações abstratas e simplificadas de uma realidade, como é o caso dos mapas de estações de metro, que representam as linhas com cores vivas e com pouca informação sobre a área física na qual o metro circula. Esta representação permite-nos transmitir uma informação objetiva e sugestiva apenas com o essencial.

Decomposição – consiste em dividir um determinado problema em partes de menor complexidade, de forma a reduzir a dificuldade do problema original, Rich et al. (2018).

Quando separamos um problema em partes menores, conseguimos compreender com mais nitidez o que está a ocorrer; ao solucionar essas partes mais pequenas e combinar as suas soluções obtém-se a resposta para o problema que era complexo. Por exemplo, quando planeamos uma viagem para um determinado destino, num determinado tempo, o que normalmente se faz é traçar uma rota e, em seguida, somar os tempos entre as diferentes escalas, desde a origem até ao destino, e fazer determinados ajustes para não ultrapassar o tempo estipulado para a viagem.

Reconhecimento de Padrões – ou identificação de padrões é reconhecer ou identificar padrões no processo de resolução de um problema e aplicar os que se revelam eficazes na resolução de problemas semelhantes. É importante reconhecer e utilizar padrões na construção e melhoria de uma solução, solicitar previsões com base em padrões identificados, procurar semelhanças e a identificação de padrões comuns a outros problemas já resolvidos de modo a aplicar, a um problema em resolução, os processos que anteriormente se tenham revelado úteis. Uma das atividades da Inteligência Artificial é o reconhecimento de padrões.

Algoritmo – é desenvolver um procedimento passo a passo para solucionar um problema de modo a que este possa ser implementado em dispositivos tecnológicos. De facto, de uma forma natural, sabemos o que é um algoritmo, pois todos os dias executamos algoritmos nas tarefas diárias, mas como o fazemos de uma forma automática não nos apercebemos. Nos algoritmos, normalmente recorremos a três tipos de instruções das ciências da computação: sequência, decisão e repetição. Uma receita de culinária é um bom exemplo para explicar o conceito. Permite-nos perceber que, se aplicar o mesmo algoritmo (sequência de instruções passo a passo), obtenho o mesmo resultado, e, caso altere essa sequência das instruções, resultados inesperados podem acontecer.

Teste e depuração – é nesta fase que se comparam os resultados reais com os resultados pretendidos. É necessário testar, encontrar e resolver problemas quando estes surgem. Desta forma, após a conclusão de um projeto ou/e de uma etapa, é necessário verificar/testar se tudo funciona como previsto, sendo, portanto, também nesta fase que muitas das vezes se propõem melhores soluções para o problema inicial, como desempenho, fiabilidade, usabilidade e acessibilidade. É importante incentivar os alunos a definirem estratégias de testagem e depuração/correção quando se identifica que algo não funciona de forma expectável ou que tem alguma imprecisão, com o intuito de encontrarem erros e melhorarem os seus processos, incentivando a sua perseverança e promovendo progressivamente a construção da sua autoconfiança.

Nos últimos anos, o PC tem sido referenciado por diversos autores, como uma competência fundamental para todos, no entanto, o debate em torno da temática, (Barr et al., 2011; Stamatios, 2022; Valente, 2016; Wing, 2006, 2014, 2017) é um construto bastante amplo na medida em que reflete, no seu conteúdo, a diversidade de conceitos, teorias, modelos, práticas ligadas à imensidão do campo das ciências da computação. O que poderá ser ainda mais lato, se acrescentarmos as inúmeras ferramentas e robôs educativos existentes no mercado, destinadas especificamente ao desenvolvimento das habilidades do PC, cada um com a sua especificidade e *Modus operandi*, compreenderemos a

complexidade do tema e as dificuldades que se colocam aos que procuram envolver-se neste tipo de trabalho.

De certo que o pensamento computacional é considerado uma competência universal, e que deve ser adicionada à capacidade analítica das crianças como ingrediente vital na sua aprendizagem, de forma a preparar os alunos para o mercado de trabalho do futuro e na compreensão dos fenómenos do mundo. Diversos países estão a promover o desenvolvimento do PC, introduzindo a área das Ciências da Computação nos currículos escolares. No entanto, permanecem dúvidas quanto à sua implementação, especificamente quais conceitos que devem ser ensinados e avaliados, quando e como.

Importa referir dois estudos recentes de 2022, por parte da Comissão Europeia e da OCDE, sobre o atual panorama da integração de programas de desenvolvimento do PC em crianças e jovens na escolaridade obrigatória.

O primeiro, divulgado pela Comissão Europeia, através do *Joint Research Centre Publication Office*, o relatório "*Reviewing Computational Thinking in Compulsory Education State of Play and Practices from the Field*" (<https://publications.jrc.ec.europa.eu/repository/handle/JRC128347>). O documento é uma versão mais atualizada e completa do estudo acerca do PC realizado em 2016 pela Comissão Europeia, aborda os avanços significativos que têm ocorrido na integração do PC na escolaridade obrigatória na Europa, entre 2016 e 2021. Para além disso, este relatório não só apresenta uma panorâmica atual daquilo que ocorre em 22 Estados-Membros da União Europeia (EU) e em 8 países não pertencentes à UE, como disponibiliza, entre outros, recomendações para decisores políticos, autoridades educativas, especialistas e professores.

O segundo, apresentado pela OCDE, no âmbito do projeto da Rede *Early Childhood Education and Care* (ECEC), disponibiliza o documento "*The state of the field of computational thinking in early childhood education*" (<https://doi.org/10.1787/3354387a-en>) da autoria de Marina Umaschi Bers, Amanda Strawhacker e Amanda Sullivan da *Tufts University*. Este documento resume a literatura empírica e teórica sobre a aprendizagem e o desenvolvimento do PC em crianças dos 3 aos 8 anos, destacando pesquisas empíricas, trabalho teórico e pedagógico e iniciativas curriculares. Apresenta, ainda, várias ferramentas e tecnologias que foram desenvolvidas na última década para apoiar o PC em crianças, incluindo atividades desligadas, ou seja, sem o uso de tecnologia. Por fim, coloca em discussão a implementação de programas de desenvolvimento do Pensamento Computacional nos países da OCDE e levanta questões sobre equidade e acesso na educação, na utilização das tecnologias.

Ambos os documentos apresentam recomendações e considerações práticas, que seguem a mesma linha de atuação, entre as quais se destacam:

- A necessidade de fortalecer a compreensão do PC como uma competência fundamental para os alunos serem cada vez mais participantes ativos no mundo digital;
- Explorar o potencial do PC para promover a resolução de problemas em diferentes domínios;
- Fortalecer as sinergias entre as partes interessadas para impulsionar a educação em computação de qualidade;
- Aumentar a conscientização sobre o propósito e os benefícios de desenvolver habilidades de PC no currículo;
- Aumentar a diversidade, o acesso, a equidade e a inclusão para a qualidade da educação em computação;
- Monitorar e pesquisar o real impacto da integração das habilidades de PC nos currículos;
- Fornecer e sustentar formação adequada ao desenvolvimento profissional dos professores, por forma a adquirir conhecimentos pedagógicos de qualidade em ciências da computação;
- Priorizar a avaliação do PC como uma competência fundamental;
- Articular uma estratégia para tecer habilidades de Pensamento Computacional no currículo;
- Integrar o Pensamento Computacional num processo contínuo desde a escola primária até ao final da escolaridade obrigatória e adequado à idade;
- Fornecer ferramentas que apoiem as crianças e jovens a serem criadores de tecnologia em vez de consumidores.

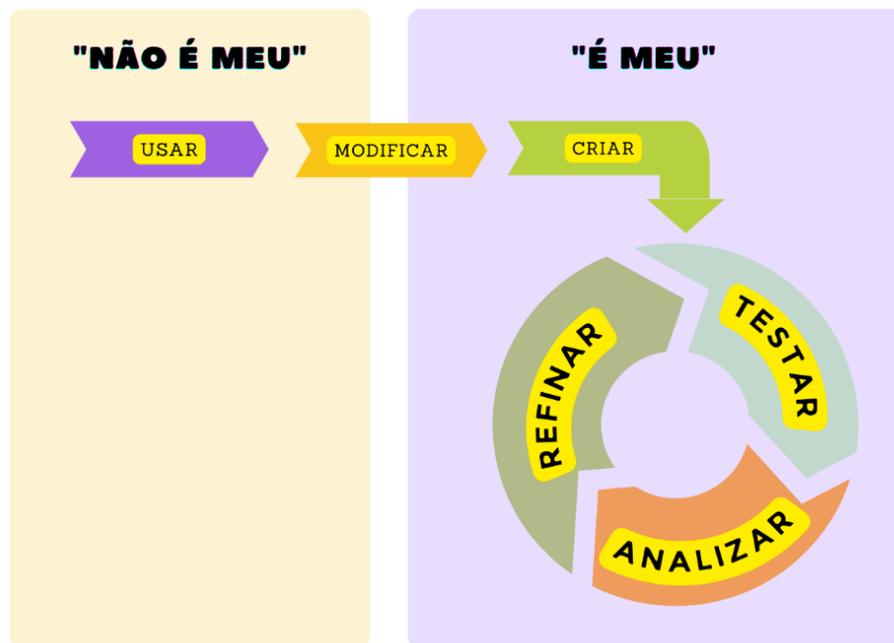
Estes dois trabalhos adquirem um especial relevo num momento em que diversos países europeus têm vindo a incluir alguns conceitos básicos das Ciências da Computação nos seus currículos, contribuindo para o desenvolvimento de competências na área do PC. Acrescente-se, ainda, o incentivo proveniente da Comissão Europeia, através do *Digital Education Action Plan 2021-2027*, onde a Educação Computacional de qualidade é considerada como elemento-chave da prioridade “Melhorar as competências digitais tendo em vista a transformação digital”.

2.2. Abordagem *Use-Modify-Create*

Para a concretização deste estudo, a utilização da Robótica Educativa necessita de ser ancorada em conceitos pedagógicos bem definidos. Deste modo, é importante definir a conceção pedagógica e a partir desta, planear a forma de intervenção seguindo uma abordagem construcionista, em detrimento de aulas instrucionais/tradicionais. Por conseguinte, foi necessária uma estratégia que gerasse suportes

essenciais para a abordagem de conceitos de PC. Assim sendo, adotou-se a estratégia de desenvolvimento curricular UMC. A progressão UMC originalmente desenvolvida por Lee et al. (2011) é uma estratégia de progressão da aprendizagem repartida por três estágios/fases e projetada para envolver intencionalmente os alunos no desenvolvimento do PC por meio de ambientes computacionais ricos.

Figura 3 – Use-Modify-Create Learning Progression



Fonte: Imagem trabalhada pelo investigador, adaptado de (Lee et al., 2011)

De uma forma resumida, o conceito ilustrado parte do princípio que um aluno execute (use) um programa ou modelo existente, passe a perceber como funciona, depois o modifique e, quando este processo estiver bastante seguro, seja capaz de criar um novo, concretizado por si, ou seja, quando já detém as competências necessárias para passar de consumidor a criador de conhecimento/artefactos.

Na fase de **usar**, é fornecido aos alunos um programa ou um modelo para explorar. Nesta fase são apenas consumidores. Os alunos têm a oportunidade de passar algum tempo a analisar como foi concebido o programa/artefacto que outros indivíduos mais competentes criaram e descobrir como funciona. Normalmente nesta fase, o professor explica o programa e conceitos necessários, faz perguntas e/ou sugestões, sendo expectável que os alunos compreendam a explicação do professor. Depois de estarem familiarizados com o programa/artefacto, os alunos estão aptos para iniciarem a modificação com base na sua compreensão.

Na fase de **modificar**, à medida que os alunos ganham confiança e estejam/se sintam cada vez mais confortáveis e familiarizados com o programa/artefacto, conceitos e contextos existentes, são incentivados a começar a alterá-lo. Nesta etapa, os alunos alteram os programas/artefactos com níveis crescentes de sofisticação, por exemplo, se for um jogo produzido no *Scratch*, a primeira coisa que um aluno normalmente faz é alterar os aspetos comuns da solução apresentada, alterando a cor ou o *sprite* e os valores iniciais das variáveis; caso seja um programa na placa Micro:Bit, ajustam o código de forma a mostrar diferentes ícones ou as iniciais do nome; já num robô, modificam o código para diferentes tipos de movimento. Nesta fase de experimentação, alteram comportamentos simples para outros mais complexos, à medida que ganham confiança com o programa/artefacto. Também, nesta etapa, o professor desempenha um papel importante ao promover sugestões de modificação intencionais de forma a provocar uma reflexão propositada sobre as ações realizadas no ambiente de programação. Esta prática permite que os participantes desenvolvam uma compreensão mais profunda. Por exemplo, quando os alunos tentam compreender o motivo pelo qual o robô não executou corretamente uma dada ação instruída no código modificado, têm de procurar o erro, fazer correções e testar novamente.

À medida que os alunos assumem níveis crescentes de desafios, ficam mais seguros para escrever os seus próprios programas. Neste momento, inicia-se a fase do **criar**, onde os alunos sentem que estão num outro nível de conhecimento, e, portanto, com confiança para aceitar novos desafios, pois já detêm as habilidades consolidadas e necessárias para passar de consumidor a criador de conhecimento/artefactos. O ciclo de criação permite aos alunos pensarem *fora da caixa* e explorar a criatividade ao trabalharem em desafios de programação/artefactos com base nas habilidades treinadas durante as etapas anteriores “uso” e “modificação” para a conclusão dos projetos. Nesta fase, o professor deve incentivar os alunos a testar, analisar e refinar as suas soluções com frequência, por forma a conseguirem uma solução mais robusta e sofisticada, (Franklin et al., 2020; Lee et al., 2011).

É importante relembrar que Usar-Modificar-Criar é um processo iterativo. O que significa que cada fase do processo deve ser continuamente preparada e, adicionando níveis crescentes de desafios/dificuldade à medida que os alunos revelem compreensão dos conteúdos. Deve garantir-se que o projeto/programa escolhido seja indicado para o nível apropriado. De acordo com Lee et al. (2011), as três fases não devem ser usadas “de forma muito rigorosa”. Existem determinadas situações em que será necessário gastar mais tempo nos primeiros estágios “usar” e “modificar” antes de chegar à fase do “criar” pois, podem não estar preparados para esse estágio. Será importante conhecer os alunos e as suas limitações.

2.3. Plataformas de Robótica Educativa

Existem inúmeras possibilidades para abordar o PC de forma a desenvolver as suas capacidades nos alunos. Há investigadores e professores que o fazem recorrendo a atividades desligadas/ *unplugged* ou a programação sem computador, outros usam as linguagens de programação por blocos e também há quem o faça aliando a programação à robótica (Valente, 2016). No entanto, cada uma das abordagens possui uma característica diferente, mas com um objetivo comum: o desenvolvimento do Pensamento Computacional.

Neste estudo, por existir um conjunto de equipamento disponível no agrupamento (robôs, placas eletrónicas ou outros artefactos tangíveis), optou-se pelo uso de uma linguagem de programação por blocos que fornece instruções aos robôs indo ao encontro do objetivo geral desta investigação. No seu artigo, Berry (2018), refere que a programação por blocos tem dados provas de uma grande eficácia para a aprendizagem da lógica da programação, sendo utilizada por alunos de várias faixas etárias, principalmente por aqueles que se estão a iniciar na aprendizagem de uma linguagem de programação. Evidencia as vantagens e desvantagens entre as linguagens de programação em texto e uma linguagem de programação visual (*Visual Programming Language*), considerando que os ambientes de programação baseados em texto são usados preferencialmente para níveis de ensino superior, favorecendo os alunos que dominam o inglês e/ou estão disponíveis para o aprender.

Com ambientes baseados em blocos, não há necessidade de decorar o vocabulário dos comandos para escrever o código. O programador pode seleccionar visualmente a partir de um conjunto de comandos, geralmente agrupados por um grupo de categorias, organizadas por cores e por tipo de instruções. As linguagens de programação visual são indicadas como as mais adequadas para o ensino da programação para os mais novos, pois permitem que os alunos se concentrem apenas na programação lógica, sem estarem preocupados em resolver os erros de sintaxe que são encontrados em linguagens tradicionais baseadas em texto (CSTA, 2016).

O Instituto de Tecnologia de Massachusetts (MIT), através de diversas experiências concretizadas no laboratório (Media Lab) em 2003, iniciou o desenvolvimento do *Scratch*, uma linguagem de programação visual baseada em blocos, que tinha como objetivo permitir que qualquer pessoa, de qualquer idade, pudesse programar. Como redigido por Mitchel Resnick, o objetivo desta linguagem é permitir que o aluno possa programar ou construir a sua história, colaborando assim no desenvolvimento da criatividade e nos processos para a resolução de problemas, Resnick et al. (2009).

Além do *Scratch*, outros projetos foram surgindo e adotaram os ambientes de programação visual por blocos com o foco no ensino da programação para crianças profundamente inspirados nos blocos de encaixar como nos brinquedos de construção LEGO.

Atualmente, existem muitas plataformas de programação baseadas em blocos, criadas por universidades e empresas de *software* que são inerentemente visuais, interativas, com recursos multimídia e com possibilidade de selecionar o idioma. Pelo facto de evitarem erros de sintaxe e problemas com as estruturas e operadores, dão uma certa confiança aos alunos para brincar, experimentar, aprender a programar e serem capazes de expressar a sua criatividade por essa via.

A maioria das plataformas de programação por blocos estão on-line e, facilmente podem ser acedidas através de um *browser*, o que permite o acesso em qualquer lugar, a qualquer momento, promovendo a partilha de projetos o que facilita o processo de aprendizagem entre pares. A programação baseada em blocos, mesmo na escolaridade inicial, é recomendada para desenvolver estratégias de resolução de problemas, planificação e processos sociais cognitivos (Stamatios, 2022).

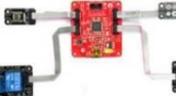
Tabela 1 - Visão geral das principais características das plataformas de programação por blocos

Nome da Plataforma	Artefacto robótico/kit	Tipo de linguagem	Estrutura/Elementos	Nível de dificuldade	Aplicabilidade
<i>Scratch</i> (https://scratch.mit.edu)	Compatível com outros artefactos, através das suas extensões	Por blocos	Movimento, Aparência, Eventos, Som, Controlo, Sensores, Atuadores, Operadores, Variáveis, Funções, Estruturas de repetição, Extensões	Fácil a Moderado	Criação de Cenários e Histórias, Jogos, Animações, Programação com artefactos robóticos.
Makecode (https://makecode.microbit.org/)	Placa Micro:Bit	Por blocos, no entanto, é possível alternar para Python e Javascript	Entrada e saídas através de sensores e atuadores (Pins), Variáveis, Som, Funções, Extensões, Arrays, Listas, Estruturas de repetição, Comunicação via Rádio, Botões de pressão	Fácil a Moderado	Programação com artefactos robóticos, Jogos, Construção de maquetes, experiências científicas
WeDo 2.0 Funciona por PC ou <i>Tablet</i>	Conjunto de peças da LEGO	Por blocos com imagens	Sequências, Sensores, Som, Led, Motores, Ciclos simples	Fácil	Controlo básico de sensores e atuadores, programação de movimentos básicos e jogos simples.
Ozoblockly (https://ozobot.com/create/ozoblockly)	Robô OZOBOT (BIT e EVO)	Por Blocos com imagens em níveis de iniciação e por blocos	Movimento, Aparência, Eventos, Som, Sensores, Led, Motores, Controlo, Sensores, Atuadores, Operadores, Variáveis, Funções, Estruturas de repetição, Arrays, Listas	Fácil a Moderado	Criação de Cenários e Histórias, Jogos, Animações.

Fonte: Tabela trabalhada pelo investigador, adaptado de <https://ducklearning.com/blogs/parent-and-educator-resources/comparison-between-different-block-base-programming-languages> consultado em 14/3/2022

Apesar do objetivo deste trabalho não passar pela descrição pormenorizada sobre as diferentes plataformas de programação por blocos existentes atualmente, será importante informar o leitor (tabela 1) sobre aquelas que, de alguma forma, suportaram a investigação e que serão alvo de detalhe nas secções (2.3.1, 2.3.2 e 2.3.3).

Figura 4 - Artefactos físicos disponíveis comercialmente

Categorization	Type of product	Examples
Packaged electronics. No programming.	1 Kits of packaged components and modules.	Snap Circuits, basic LittleBits, Circuit Stickers  
	2 Packaged programmable products (not boards). Programmable via PC or phone. Often battery powered.	Robot turtles. Sphero, Ozobot, Kibo, Dash and Dot, BeeBot, Cubetto      
Board level peripheral devices. Need PC during use.	3 Integrated I/O devices for PCs.	Makey-makey, PicoBoard, BlinkM, Sense Board   
	Modular I/O devices for PCs.	Phidgets 
Board level embedded devices. Need PC to program but operate standalone. Can be battery powered.	4 Microcontroller boards with integrated I/O devices.	micro:bit, Light Blue Bean, Arduino Esplora, Circuit Playground, Calliope   
	Microcontroller boards with low-level I/O.	Crumble, BASIC stamp, ARM mbed, Chibi Chip    
	Microcontroller boards with support for modular I/O.	Arduino variants   
		.NET Gadgeteer, TinkerKit, Hummingbird   
Board level general-purpose devices. Often use wired power.	5 Often used without PC. I/O available through accessories.	Raspberry Pi, BeagleBone, Intel Galileo   

Fonte: Imagem trabalhada pelo investigador, com base na classificação de (Hodges et al., 2020)

Para além das plataformas de programação por blocos, o uso de diferentes contextos, como é o caso do recurso a artefactos físicos para o ensino de programação, pode inspirar o interesse dos alunos e em alguns contextos parecem ser mais motivadores do que outros e proporcionar a possibilidade de resolver problemas, contextualizados, parecidos com o mundo real, de uma forma criativa, Garneli et al. (2015). Existem muitos tipos diferentes de dispositivos físicos e robôs, conforme ilustrado na Figura 4. Os dispositivos podem incluir componentes eletrónicos sem necessidade de programação, robôs programáveis e conjuntos para construção, placas programáveis com dispositivos de *input* e *output* integrados ou externos que precisam de um computador durante a operação, placas programáveis incorporadas alimentadas por baterias, que podem funcionar sem computador, e placas programáveis comuns com interligação via cabo, Hodges et al. (2020).

Muitas outras plataformas poderiam ser assinaladas, no entanto, tendo em conta a especificidade desta investigação, as opções apresentadas são para a escolaridade inicial, nomeadamente aquelas que estão interligadas com os conjuntos robóticos explorados neste estudo (LEGO WeDo 2.0, Micro:Bit e Ozobot). De acordo com o autor, a figura anterior representa os dispositivos físicos (placas e robôs) classificados não por idade, habilidade ou conceitos específicos relacionados com o PC, mas por uma taxonomia simples de agrupamento de produtos com fatores, formas e recursos técnicos semelhantes.

Na categoria 1 e 2 estão classificados por uma variedade de conjuntos modulares e brinquedos programáveis muito populares, normalmente com um custo elevado como é o caso dos conjuntos de construção da LEGO, Ozobot e Kibo. No entanto, nos últimos anos temos assistido uma grande variedade de dispositivos de baixo custo (categorias 3, 4 e 5), logo, ao alcance de um grande número de escolas e, sem dúvida, que são os grandes impulsionadores da massificação e adoção da programação em artefactos físicos. Estas placas controladoras permitem a interligação com componentes eletrónicos como sensores e atuadores ou módulos de ligação disponibilizados por diversos conjuntos como, por exemplo, este conjunto de componentes eletrónicos que permite novas ideias e projetos (<https://www.botnroll.com/pt/bbc-microbit/3957-kit-37-sensores-e-atuadores-para-micro-bit-keyestudio-ks0361.html>).

Na categoria 5, encontram-se diversos dispositivos que podemos considerar como minicomputadores, por terem uma capacidade de processamento maior comparativamente com os anteriores. Desta forma, fornecem maior flexibilidade, mas são mais complexos e requerem um conhecimento mais profundo do utilizador, quer a nível de programação quer em termos de conhecimentos de eletrónica/*hardware*.

Os conjuntos robóticos utilizados para a realização desta investigação foram os modelos assinalados na figura anterior (A – Robô Ozobot, B – Conjunto LEGO WeDo 2.0 e C - Placa microcontroladora Micro:Bit) categorias 2 e 4.

A escolha destes recursos para a realização desta investigação foi acordada pelo investigador e pelas professoras cooperantes responsáveis pela lecionação da disciplina de OCPR, de acordo com as seguintes razões:

- Robô Ozobot e Placa Micro:Bit – o departamento adquiriu no ano letivo 2021/2022 dois conjuntos de 12 Ozobot evo classroom kit (24 robôs) e 24 placas Micro:Bit um número suficiente para trabalhar de forma individual ou em grupo;
- Os conjuntos assinalados anteriormente não foram trabalhados, devido às restrições do COVID-19 e a alguma falta de conhecimento do professor responsável na altura;
- existência de 24 caixas completas LEGO WeDo 2.0 e conhecimento técnico e pedagógico, resultante da experiência desde o ano letivo 2018/2019;
- Conhecimento prévio por parte do investigador das tecnologias enumeradas anteriormente.

Obviamente que as razões enumeradas anteriormente são apenas circunstanciais e não justificam o propósito desta investigação, contudo, foram identificados diversos investigadores que nos seus trabalhos fazem referência à RE como uma das várias estratégias para disseminar ou promover os conceitos relacionados com o desenvolvimento do pensamento computacional e consequentemente fazem menção a estes artefactos robóticos, (Bakala et al., 2021; Bers et al., 2022; Fantinati & Rosa, 2021; Guedes et al., 2021; Neto & Bertagnolli, 2021). Naturalmente, sendo um dos objetivos desta investigação a caracterização dos materiais pedagógicos que suportam as atividades, é expectável, segundo as investigações anteriores que a RE promoverá um contributo para o desenvolvimento computacional na escolaridade inicial.

2.3.1. LEGO WeDo 2.0

Dado ser um dos artefactos robóticos, com os quais os alunos foram solicitados a desenvolver alguns dos seus projetos, bem como sendo um dos objetivos deste estudo a caracterização de materiais pedagógicos que alicerçam as atividades, assim o investigador equacionou a importância da inclusão de uma breve descrição e caracterização do conjunto robótico LEGO WeDo 2.0.

O conjunto LEGO Education WeDo é o resultado de uma colaboração entre Mitchel Resnick, na altura responsável pelo projeto *Lifelong Kindergarten* no MIT Media Lab e Erik Hansen, responsável pelo

desenvolvimento de recursos eletrônicos do grupo LEGO. O conjunto de construção da LEGO é destinado a crianças dos 7 aos 9 anos e foi lançado em 2009 na versão WeDo 1.0 e, mais tarde, em 2016 sofreu um *upgrade* para a versão 2. Em traços gerais, o conjunto robótico apresenta várias peças LEGO, com um motor, dois sensores e uma placa microcontroladora (*Smarthub*) para ligação a um computador. O *software* de programação visual por blocos é bastante simples, basta apenas arrastar e largar, uma abordagem destinada a crianças, que assim aprendem divertidamente, de forma inspiradora e envolvente, enquanto aprendem conceitos de programação e desenvolvem habilidades motoras e cognitivas através da construção de robôs. O conjunto WeDo 2.0 é constituído por 280 peças LEGO, com um suporte para as peças dividido em 13 secções por forma a classificar as peças por funcionalidade e tamanho, sendo as peças maiores são armazenadas na secção por baixo do suporte. Este conjunto permite o encaixe de peças LEGO tradicionais e vem equipado com diferentes tipos de engrenagens, ligações esféricas, vários tipos de eixos, elásticos e outros, que potenciam a construção de diversos tipos de artefactos nos diferentes temas, como animais, transportes, robôs e muitos outros, dependendo da criatividade e engenho que poderá incentivar e motivar crianças que tecnologicamente são inseguras a gostarem desta área. Para além das peças LEGO, o conjunto LEGO WeDo 2.0 contém 4 equipamentos eletrónicos; um motor de tamanho médio, um sensor de inclinação, um sensor de movimento e um microcontrolador. A comunicação é feita via Bluetooth a um dispositivo eletrónico (*tablet* ou computador), sendo possível controlar o motor e ler os sensores.

Figura 5 - Conjunto LEGO WeDo 2.0



Fonte: Imagem retirada pelo investigador, de <https://education.lego.com/en-us/products/lego-education-wedo-2-0-core-set/45300#wedo-20> em 4/7/2022
Descrevendo detalhadamente temos:

- WeDo 2.0 Smarthub – como já foi referido é o bloco de construção que contém a base do sistema eletrónico, que pode ser alimentado por duas pilhas (alcalinas ou recarregáveis) ou ainda, por um módulo *Power Bank* recarregável, para além das funções descritas anteriormente

contém um led que pode ser programável com 10 cores diferentes e um botão verde que permite a ligação ao *software* ou App, via *Bluetooth*. Possui, ainda, duas portas para a ligação aos motores externos e sensores, o *Smarthub* também permite trabalhar com outros microcontroladores de forma colaborativa, por exemplo, criar um carro com 2 *Smarthub* e 4 motores;

- Motor - o Motor Médio pode ser programado no sentido horário e anti-horário e mover-se em diferentes níveis de potência, com um design que permite a ligação com outras peças LEGO.
- Sensor de movimento - é um sensor infravermelho ligado ao *Smart Hub* que pode detetar objetos num alcance de 15 cm, o que permite ser utilizado em projetos para medir distâncias dos objetos à sua frente ou detetar movimento;
- Sensor de inclinação - o sensor de inclinação informa a direção, se inclinado. Uma nova funcionalidade no sensor *Tilt WeDo 2.0* é o modo de "vibração". O sensor de inclinação deteta seis posições diferentes (0,3,5,7 e 9).

Em termos de programação, o LEGO WeDo 2.0 necessita do software gratuito que pode ser retirado da página oficial (<https://education.lego.com/en-us/downloads/retiredproducts/wedo-2/software>), para diferentes dispositivos e sistemas operativos, computador (Windows e/ou Mac OS), *tablets* (Android, Chromebook e iOS). Uma das vantagens é a possibilidade de usar *tablets* na programação do conjunto WeDo 2.0 que o torna indicado para trabalhar com a interface de programação em ecrãs sensíveis ao toque, com os quais as crianças se encontram familiarizadas.

Figura 6 - Software WeDo 2.0 - Interface em modo Tablet



Fonte: Imagem trabalhada pelo investigador, retirada do *Software WeDo 2.0 - Interface em modo tablet* (<https://thecodingfun.com/wedo-2-0-software-interface-example/>)

O painel de bloco de codificação apresentado na figura anterior, permite deslocar para a esquerda e direita e arrastar o bloco pretendido para a área de programação.

Os blocos de programação usam pequenas imagens sugestivas e informativas da função de cada bloco, em vez de texto. Por exemplo, para mover um motor, o aluno necessita de arrastar e soltar um ícone “*motor turn*” na área de trabalho.

Parece ser uma solução mais acolhedora e menos intimidadora para alunos que estão a iniciar a programação em robótica, alunos com dificuldades de leitura ou alunos que ainda não aprenderam a ler nem a escrever.

Os blocos são unidos linearmente de forma horizontal para que as crianças consigam perceber facilmente a sequência do código, estando agrupados por cores de acordo com a sua funcionalidade, conforme a imagem seguinte apresentada.

Figura 7 - Blocos de Programação LEGO WeDo 2.0



Fonte: Imagem retirada pelo investigador, da informação disponibilizada pela LEGO Education WeDo 2.0

1 – Blocos funções de fluxo – destinados a controlar a sequência de programação, contêm o bloco início, bloco ao pressionar uma tecla, bloco iniciar uma mensagem, bloco enviar uma mensagem, bloco de espera por..., bloco tocar música e bloco repetir.

2 – Blocos funções de Motor – com o objetivo de controlar a ação dos motores, contêm o bloco virar no sentido anti-horário, virar no sentido horário, bloco para controle da potência do motor, desligar motor, manter motor ligado por e bloco para definição da cor led do *Smarthub*.

3 – Blocos funções de monitor/visor – destinados à visualização de números, operações, imagens e redimensionamento do ecrã, contêm o bloco monitor (irá apresentar valores no ecrã), o bloco imagem de fundo do ecrã, bloco ecrã tamanho grande, bloco ecrã tamanho médio, bloco fechar bloco monitor, bloco adicionar ao monitor, bloco retirar ao monitor, bloco multiplicar pelo valor apresentado monitor e dividir pelo valor do monitor.

4 – Blocos dos sensores – destinados a controlar a leitura dos sensores (movimento e inclinação), contêm o bloco ler valor do sensor infravermelho (distância), bloco de alteração de distância mais próxima, bloco de alteração distância mais longe, bloco de qualquer alteração de distância, bloco inclinar para baixo, para cima, para a direita, para a esquerda e bloco de qualquer inclinação.

5 – Blocos de entrada – com o objetivo de controlar as mudanças nos blocos de entrada, bloco de entrada de texto, entrada numérica, entrada do monitor/visor, entrada aleatória e entrada do sensor de som.

6 – Outros Blocos – O bloco caixa de diálogo.

O conjunto LEGO WeDo 2.0 e o *currículo* LEGO Education WeDo 2.0 foram projetados para o primeiro ciclo do ensino básico e colocam como meta principal o despertar do interesse das crianças pela aprendizagem das áreas da ciência, engenharia, tecnologia e programação, recorrendo à robótica e à programação simples, com base nos padrões de ciência e tecnologia da próxima geração e oferece conteúdo científico fundamental para alunos da escolaridade inicial (Lego, 2017).

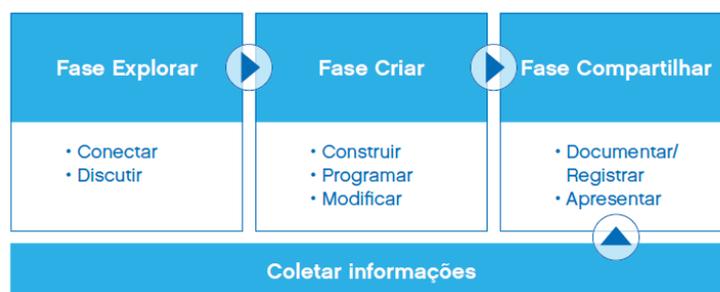
As diversas atividades práticas apresentadas no *currículo*, estimulam a curiosidade dos alunos, enquanto envolve os seus conhecimentos na ciência, engenharia, tecnologia e programação, desenvolvendo o pensamento crítico e resolução de problemas. O LEGO WeDo 2.0 é uma solução de aprendizagem pela investigação que fornece aos alunos a confiança para fazerem perguntas e as ferramentas para encontrarem as respostas e resolverem problemas do quotidiano. A exploração dos projetos não fornece aos alunos tudo que eles precisam saber, pelo contrário, incentiva-os a questionar o que sabem e a explorar o que ainda não entendem (Lego, 2017).

O LEGO WeDo 2.0 possui uma variedade de projetos, que se encontram organizados da seguinte forma:

- 1 projeto introdutório, dividido em 4 partes para trabalhar com as funções básicas do LEGO WeDo 2.0;
- 8 projetos curriculares orientados, com instruções passo a passo, para o projeto completo;
- 8 projetos curriculares livres, com uma possibilidade de trabalho mais ampla.

Os 16 projetos (8 orientados e 8 livres) estão divididos em três fases: a fase explorar, a fase criar e a fase compartilhar.

Figura 8 - As três fases da metodologia LEGO Education WeDo 2.0



Fonte: Imagem retirada pelo investigador, da informação disponibilizada pela LEGO Education WeDo 2.0
[https://education.lego.com/en-gb/product-resources/wedo-2/teacher-resources/teacher-guides#english-\(us\)](https://education.lego.com/en-gb/product-resources/wedo-2/teacher-resources/teacher-guides#english-(us))

O *software* WeDo 2.0, permite aceder:

- Ao pacote curricular definido pela LEGO Education;
- A projetos de introdução, guiados e abertos;

- Às instruções de construção, em formato de vídeo e orientações na programação;
- A uma biblioteca de modelos já construídos e programados.

Cada unidade fornece um resumo inicial como ponto de partida. As instruções permitem respostas diferenciadas e uma variedade de soluções criativas. O papel do professor é proporcionar aos alunos ferramentas e autonomia, por forma a criar soluções que vão ao encontro das necessidades de cada aluno e não fornecer conhecimento pronto.

Os conjuntos LEGO WeDo 2.0 também podem ser manipulados por outro *software*, como é o exemplo do *Scratch* através de uma extensão permitindo, desta forma, outras opções de codificação com um grau superior de complexidade.

Da leitura efetuada e dos resultados escrutinados dos estudos realizados com o conjunto robótico LEGO WeDo 2.0, (Chalmers, 2018; Demetriadis et al., 2012; Üşengül & Bahçeci, 2020), foi relatado que os alunos aparentam desenvolver habilidades de pensamento computacional, nomeadamente sequências, ciclos, reconhecimento de padrões, depuração e resolução de problemas.

Dos diferentes estudos, os professores também observaram que os alunos mostraram competências no desenvolvimento dos níveis de PC na escolaridade inicial. O recurso à robótica educativa é uma metodologia de ensino que promove o desenvolvimento cognitivo e o desenvolvimento emocional, Chalmers (2018).

Em suma, o conjunto LEGO WeDo 2.0 evidencia ser um bom instrumento para introdução aos conceitos das ciências da computação e do PC, por permitir projetos baseados na resolução de problemas. Neste sentido, torna-se importante ajudar as crianças a desenvolver um conjunto de habilidades na área do PC, por forma a estarem mais preparadas para a sociedade de amanhã, os conceitos devem ser transmitidos, passo a passo e com uma sequência lógica. Caso contrário, podemos cair no erro de desenvolver atividades pontuais e/ou apenas porque as crianças acham divertidas, em vez de um currículo com uma sustentação pedagógica e científica.

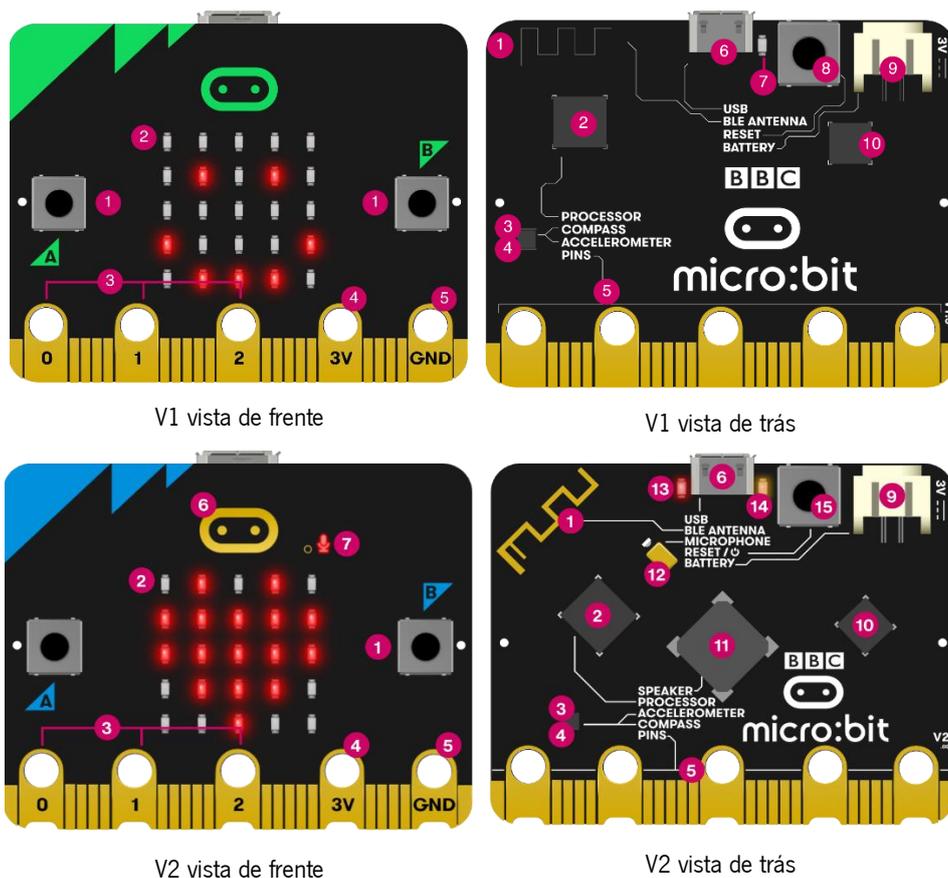
2.3.2. Placa Micro:Bit

O Micro:Bit é um pequeno microcontrolador (computador de bolso), programável com múltiplos recursos, e de fácil utilização. A British Broadcasting Corporation (BBC) criou em parceria com mais de 25 organizações espalhadas pelo mundo, uma pequena placa de *hardware* programável para as crianças dos 7-12 anos do Reino Unido (BBC, 2022).

A Micro:Bit Foundation, divulga desde setembro de 2016, um projeto da BBC *Make It Digital*, que já distribuiu cerca um milhão de placas Micro:Bit nas escolas do Reino Unido, a crianças em idade escolar. Este recurso foi construído com o objetivo de criar uma experiência divertida e de fácil utilização, inspirando todas as crianças a adquirir conceitos básicos das ciências da computação que promovam a criatividade, o desenvolvimento de competências e habilidades de pensamento computacional (BBC,2022) . O Micro:Bit é um dispositivo de computação física que fornece uma ligação entre conceitos abstratos e experiências claras para o aluno, (Hodges et al., 2020; Sentance et al., 2017).

Existem dois modelos da placa Micro:Bit (figura 9): a versão original (V1) e a versão com inclusão de som (V2). É uma placa de tamanho reduzido, com baixo consumo e compacta que incorpora muito *hardware*. Na parte frontal, disponibiliza 25 pequenas luzes vermelhas (LEDs) (2), dispostas numa matriz (5x5) que permitem a exibição de mensagens, ícones, números e palavras. Dois botões A e B (1) programáveis que podem ser usados na interação com os programas. Apresenta na parte inferior os *Pins*-GPIO (3) de ligação a componentes externos (sensores, motores e outros). São 25 pinos e cinco ligações em anel (três para entradas/saídas digitais/analógicas, um para energia [3V] (4) e um para a terra [GND] (5), informação disponibilizada em (<https://makecode.microbit.org/device/pins>).

Figura 9 – Constituição da placa Micro:Bit V1 e V2



Fonte: Imagens trabalhada pelo investigador, retiradas de <https://microbit.org/get-started/user-guide/overview/>

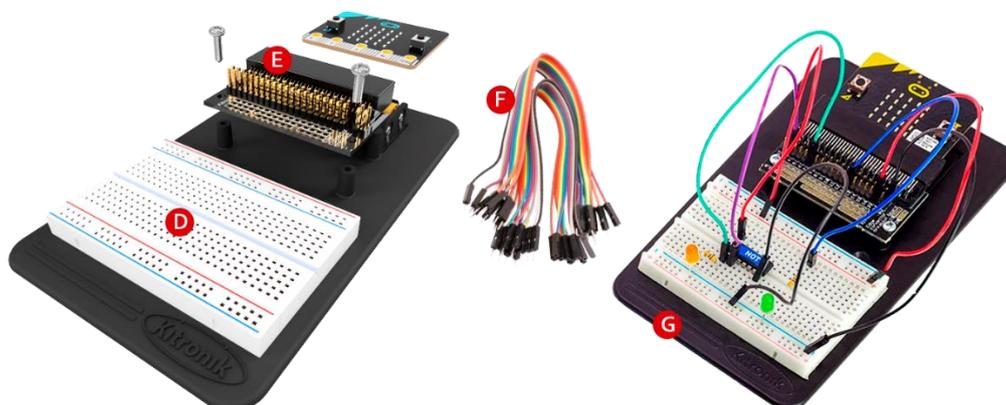
A grande vantagem das ligações em anel é que permitem com facilidade usar pinças de crocodilo para uma rápida prototipagem com sensores externos e atuadores. A versão mais recente da placa incorpora um microfone (7) e um sensor de toque (6), conhecido por logótipo táctil, entre outros *upgrades* importantes elevando a versatilidade e potencialidade desta placa (<https://kitronik.co.uk/blogs/resources/explore-micro-bit-v1-microbit-v2-differences>).

O dispositivo incorpora também na parte de trás, uma antena *Bluetooth Smart* (1) para interagir com outras placas Micro:Bit e outros dispositivos, um microcontrolador (2) ARM Cortex-M0 (nRF51822) de 16 *bits* e 16MHz, com 256KB Flash e 16KB de RAM, na versão 1 e na versão dois um processador mais robusto, um nRF52833 de 32 bits e 64MHz, com 512KB Flash e 32KB de RAM. A placa integra um sensor de temperatura (2), um acelerómetro (3), uma bússola (4), bem como um conector microUSB (6) (alimentação e transmissão de dados) e JST (9) de dois pinos para diferentes opções de alimentação, como é o caso da alimentação através de pilhas (2 x 1,5 V). Muito útil, em projetos que requerem mobilidade da placa.

Para a utilização do dispositivo é necessário ligar a placa a um computador ou *tablet*, através de um cabo microUSB, que permitirá por um lado a alimentação da placa e por outro a transferência de ficheiros do computador/ *tablet* para a placa.

Este dispositivo permite trabalhar conjuntamente com outros recursos educativos, alargando a aprendizagem de conceitos como o de pequenos circuitos eléctricos/eletrónicos ou até desenvolver projetos mais complexos. A título de exemplo, apresenta-se uma solução básica (figura 10) usada neste estudo, que inclui uma placa de ensaio (*breadboard*) (D), componentes básicos de eletrónica (Leds) e fios de ligação (F).

Figura 10 - Kit básico de eletrónica



Fonte: Imagens trabalhadas pelo investigador e retiradas de <https://kitronik.co.uk/>

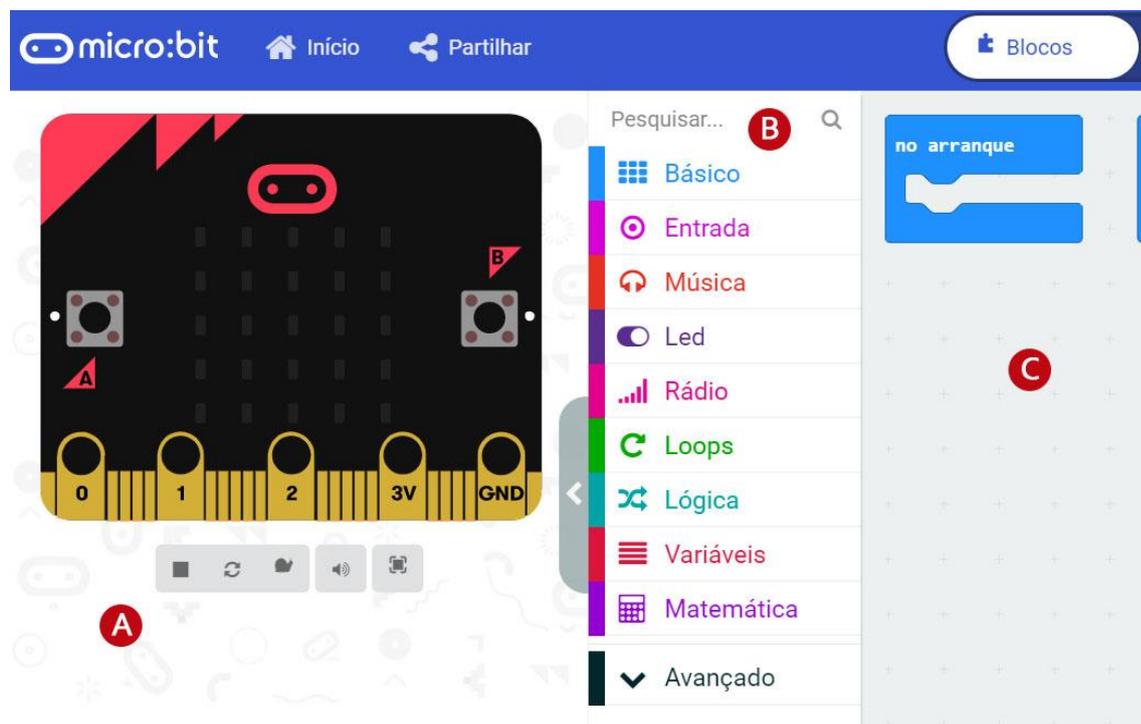
Importa referir que a placa Micro:Bit é programada através da plataforma MakeCode, disponibilizada pela Microsoft, (<https://www.microsoft.com/en-us/makecode>) que fornece um conjunto de bibliotecas, aulas

e atividades interativas, que permite aos alunos adquirirem e desenvolverem atividades no âmbito das ciências da computação.

A plataforma Makecode em conjunto com a placa Micro:Bit, possibilitam um conjunto de recursos capazes de produzir soluções inovadoras e criar experiências enriquecedoras no mundo das ciências e até nos conceitos ligados à realidade. Potencia a curiosidade e auxilia na exploração e aquisição de conhecimento através da experimentação, contextualizando os conceitos teóricos em construções e mecanismos concretos, Rogers & Siever (2018).

O editor MakeCode (<https://makecode.microbit.org/>) é *online* e inclui um simulador interativo, fornecendo aos alunos um *feedback* imediato sobre as execuções das ações programadas antes deste ser enviado para a placa física, o que facilita o teste e a depuração do código.

Figura 11 – Editor Makecode para Micro:Bit



Fonte: Imagem trabalhada pelo investigador, retirada do editor MakeCode

No lado esquerdo da *interface* (A), encontra-se o simulador completo de uma placa Micro:Bit real. Poderá ser usado durante a escrita do programa para testar a evolução do código. O simulador é completamente interativo, ou seja, é possível clicar com o rato nos botões ou agitar a placa Micro:Bit no simulador que reage de acordo com o que está programado, como se se tratasse de uma placa física. Um recurso importante para escolas que não tenham orçamento para adquirir as placas físicas.

O editor (figura 11) contém um conjunto de blocos (B), classificados por cores, de acordo com o tipo de instruções, muito semelhante a outras plataformas amplamente conhecidas, como o *Scratch*. Também

suporta programação orientada a eventos, como os sinais enviados por sensores (botões, temperatura, som, entre outros) e a execução em simultâneo de vários blocos.

Na parte direita da *interface* (C), situa-se a área de programação, local onde os blocos de programação são arrastados e organizados, quando um bloco evidenciar uma cor viva, significa que a placa Micro:Bit é capaz de o processar.

O uso de blocos potencia uma semântica mais transparente para o aluno, permitindo adquirir um conjunto de instruções básicas de programação orientada a eventos, desde uma simples expressão à construção de tarefas mais complexas, apoiadas em recursos que ajudam os jovens programadores a evitar erros de principiante.

As atividades propostas nesta investigação, com a placa Micro:Bit (secção 3.5.1, sessões de 3 a 8), foram concretizadas tendo como linha de pensamento o trabalho realizado pelos investigadores, Kiang & Kiang (2019), nomeadamente nas unidades 3, 4 e 5, (<https://makecode.microbit.org/courses/csintro>) relacionadas com os conceitos básicos de ciências de computação (variáveis, condições e iterações), alinhado com os objetivos e conceitos de aprendizagem definidos pela CSTA. Este trabalho faz parte de uma série de cursos disponibilizados pela plataforma MakeCode, para alunos e professores e outro público diversificado.

Os autores propõem uma abordagem diferente para ensinar os conceitos básicos das ciências da computação, estimulando a curiosidade, através de trabalhos e projetos centrados no movimento *MAKER*, captando a envolvência dos alunos, por forma a atrair tanto alunos como alunas. Um curriculum, centrado no aluno, como criador e mentor de soluções capazes de resolver problemas, ou seja, todos os programas são criados para resolver um problema ou servir a um propósito. Esta estratégia identificada pelos autores, leva os alunos a mobilizarem os conceitos de programação na implementação de projetos criativos (e muitas vezes colaborativos), reconhecendo a importância das habilidades da programação na construção de soluções.

No inquérito por *focus group* (secção 4.2) as professoras cooperantes reportaram que os alunos gostaram da experiência da utilização da placa Micro:Bit nas diferentes atividades propostas. Sendo assim, a sua utilização foi o instrumento primordial da investigação.

2.3.3. Robô Ozobot

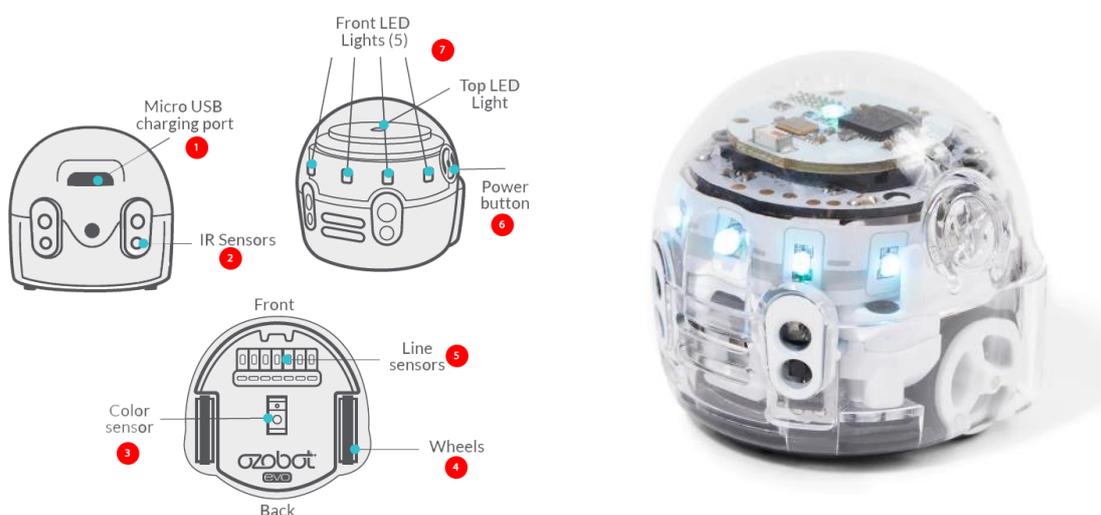
Para além dos artefactos robóticos da LEGO WeDo 2.0. e da placa Micro:Bit, foi utilizado neste estudo o robô Ozobot Evo (<https://ozobot.com/>) em duas sessões de trabalho. A escolha deste robô recai na sua facilidade de utilização e funcionamento, podendo ser programado com ou sem computadores,

facilitando a transição da programação com atividades desligadas ou *unplugged*, para a programação com computadores. Um outro aspecto evidente é o seu tamanho reduzido, que permite transportá-lo, o que facilita no manuseamento das crianças com uma faixa etária mais baixa, que é o caso deste estudo. O Ozobot é um pequeno robô esférico desenvolvido e fabricado pela norte-americana Evolve, que ajuda as crianças na aquisição de conceitos básicos de programação e a raciocinar dedutivamente, muito importante no desenvolvimento do pensamento computacional. De acordo com (Körber et al., 2021; Žáček & Smolka, 2019), que utilizaram o robô Ozobot nas suas investigações, consideram-no adequado à introdução de conceitos de programação em crianças na escolaridade inicial, com o objetivo de promover as dimensões do pensamento computacional.

Este robô apresenta uma desvantagem relativamente ao conjunto LEGO WeDo 2.0 e placa Micro:Bit, é o facto de não permitir a montagem do robô, tarefa que as crianças também adoram e por esse motivo condiciona a criatividade e, por outro lado o seu tamanho reduzido limita a personalização do robô com materiais decorativos e que pode, acidentalmente, cobrir alguns dos sensores. Outro aspeto menos positivo, fruto da sua dimensão, prende-se com a duração da bateria, que dependendo da sua utilização e da constante ligação a um dispositivo via *Bluetooth*, causa uma drenagem da bateria.

Apesar das desvantagens enunciadas anteriormente, é um robô robusto, compacto, com um design arrojado e de fácil utilização com um grande potencial para ensinar conceitos das ciências da computação (simples ou complexos), que se adaptam a diferentes faixas etárias do pré-escolar ao secundário. Impressiona pelo número e qualidade dos componentes eletrónicos que são visíveis, dada a transparência física do robô.

Figura 12 - Robô Ozobot Evo - Hardware



Fonte: Imagem trabalhada pelo investigador, retirada de <https://shop.ozobot.com/>

Em termos de *hardware*, o Ozobot é constituído por seis conjuntos de LED (7), cinco na parte frontal e um em cima, uma saída de som, quatro sensores de proximidade (2), sensores de cores (3), sensores de linha (5), um botão *on/off* (6) e uma antena *Bluetooth smart*, aumentando a possibilidade de ligação com outros dispositivos e dois motores (4) acionados separadamente. Todos os sensores e atuadores são programados, possibilitando inúmeras ideias e cenários, fomentando a criatividade e o interesse dos alunos.

Sem dúvida, que o Ozobot tem um potencial de programação considerável, pois pode ser programado em diferentes linguagens (visuais por blocos, Python ou C++), com diferentes níveis de dificuldade, com ou sem computadores.

Figura 13 - Atividade com Ozobot, codificação por cores



Fonte: Imagem trabalhada pelo investigador

Uma das estratégias é a programação representada por código de cores (imagem anterior) sem o recurso a computadores ou *tablet*, em que o robô segue caminhos desenhados com cores que controlam a velocidade e a direção. O Ozobot funciona bem em superfícies tradicionais como mesas, papel de cenário ou outro tipo de papel com superfície lisa. Dependendo da sequência de cores reconhecida pelo sensor de cores, o Ozobot executa uma determinada ação de acordo com um determinado código de cores (<https://ozobot.com/create/color-codes>). Este recurso simples é uma das estratégias que poderá ser implementada nos estados iniciais da aprendizagem de conceitos de programação, pois permite que os alunos pratiquem conceitos básicos de uma forma divertida e sem recurso a computadores.

Outra forma de programar o pequeno robô Ozobot é a programação por blocos através da aplicação OzoBlockly (<https://ozoblockly.com/>), um recurso totalmente gratuito acedido através de um navegador Web, caracteriza-se por ser intuitivo, bem desenhado/desenvolvido, completo com inúmeras ferramentas de fácil utilização, adaptando-se aos diferentes níveis de idade. Funciona em diferentes dispositivos

eletrônicos (computadores, *tablet* ou *smartphone*) e é destinado à programação dos robôs Ozobot Bit e Evo, incorporando uma nova funcionalidade, o MetaBot (<https://ozobot.com/create/metabot>) destinado a projetar o robô Ozobot no campo da realidade aumentada.

Figura 14 - Plataforma de programação - OzoBlockly



Fonte: Imagem trabalhada pelo investigador, retirada de <https://ozoblockly.com/editor>

O ambiente de programação do OzoBlockly está classificado em cinco níveis de dificuldade (1), desde o nível um, para alunos com menos experiência ou mais novatos, até ao nível cinco para alunos mais experientes e com outra maturidade. Fornece também programas exemplo bem documentados, para que o aluno teste e tente perceber como funciona, depois o modifique e, quando este processo estiver bastante seguro, seja capaz de criar um novo que é a essência da abordagem UMC, descrita na secção 2.2 deste trabalho.

No modo 1 – *Pre-Reader*, como o nome sugere, é um nível básico, destinado a crianças que ainda não sabem ler. Pode ser usado no pré-escolar e alunos do 1.º e 2.º anos do ensino básico. Os blocos de programação são apresentados em forma pictográfica, sendo o robô programado em tarefas básicas como andar para a frente, girar, usar efeitos de luz e som pré-existentes ou tempos de espera em segundos.

O modo seguinte, designado por *Beginner*, introduz o conceito de ciclo (*loop*) e altera a aparência dos blocos de programação, incluindo texto descritivo e imagens mais pequenas. O módulo 3 – *Intermediate*, inclui instruções para leitura dos sensores e estruturas condicionais, permitindo uma interação com o

robô Ozobot. O modo seguinte, o modo 4 – *Advanced*, deixa de ter ícones nos blocos de programação, e permite criar funções e variáveis. Neste nível estão presentes blocos de conceitos matemáticos e físicos, como os números aleatórios, graus, ângulos e operações. Por último, o modo 5 – *Master*, vem acrescentar novas opções relacionadas com as ciências de computação como listas, *arrays* e a programação do botão do Ozobot.

Alunos com conhecimentos em Matemática e programação, têm com este recurso uma abundância de ferramentas para criar projetos e testar hipóteses, incluindo a oportunidade de programar em *Python* (<https://www.python.org/>), uma das linguagens de programação mais utilizadas no mundo, através do editor Ozobot Python editor. Em suma, o OzoBlockly é um recurso muito bom para adquirir conhecimentos das ciências de computação, no entanto, a progressão dos alunos deve ser acompanhada e supervisionada, porque é bem provável que o aluno se perca nos diferentes estágios, guias, exemplos e blocos de código.

O robô Ozobot Evo e a plataforma OzoBlockly têm as condições necessárias para ajudar as crianças e jovens na aprendizagem de conceitos das ciências da computação, desde conceitos mais simples até aos mais avançados, mas como em tantas outras situações, pode ser um desafio usá-los sem qualquer orientação adicional por parte de um professor. De referir que a plataforma OzoBlockly não está disponível em língua portuguesa, pelo que poderá ser uma dificuldade acrescida, principalmente nos primeiros anos.

Para além das atividades descritas anteriormente, está disponível para os mais novos uma aprendizagem através da gamificação (<https://ozobot.com/create/challenges>), onde os alunos são convidados a resolver desafios que dão pontos. É uma boa solução para as crianças que querem descobrir o potencial do robô, por sua iniciativa e necessitam de alguma motivação extra. Um outro recurso disponível para quem não tem o robô físico é o simulador (<https://games.ozoblockly.com/shapetracer-freeform>), que, essencialmente, permite simular e executar programas, com um ambiente muito semelhante ao OzoBlockly. Esta solução é muito útil quando os alunos estão em ensino remoto como foi o caso dos dois últimos anos, por motivos da COVID-19 e, obviamente pelo custo do robô.

Está disponível também uma aplicação Evo App (<https://ozobot-homepage.appspot.com/play/apps>) para *tablet* e *smartphone*, que oferece o básico da plataforma OzoBlockly, através do *ShapeTracer*, o *OzoLaunch*, entre outros jogos. Depois de instalar a aplicação e estabelecer uma ligação ao pequeno robô Ozobot, através do *Bluetooth*, é possível controlar de forma livre o robô, controlar sensores, programar, usar os códigos de cores, partilhar projetos ou simplesmente por pura diversão.

Outra observação prende-se com o enorme conjunto de recursos pedagógicos disponibilizados na plataforma para professores e alunos, como é o caso do Ozobot Classroom (<https://ozobot.com/educate/classroom>), uma plataforma online premiada e que oferece aos professores o acesso a múltiplos recursos, com a possibilidade de formação, *webinars* e treino interativo. Cada aula contém uma planificação cuidada, um vídeo explicativo e instrutivo, adequado ao aluno, possibilitando o uso em sala de aula ou remotamente. Todas as aulas foram testadas e validadas na prática, estando alinhadas com os objetivos e conceitos de aprendizagem definidos por diversos padrões como a CSTA e o ISTE, por forma a garantir que os alunos estão a adquirir competências e aptidões do século XXI que possivelmente irão necessitar no futuro.

As aulas são projetadas para todos, com ou sem experiência nas ciências da computação. A plataforma *Ozobot Classroom* oferece aos professores e às escolas *insights* de engajamento, mesmo quando os alunos aprendem sem recurso a computadores com aulas práticas baseadas em projetos. É um lugar ideal para aprender como funciona o robô Ozobot, assim como adquirir os conceitos iniciais sobre ciências da computação, Matemática, ciências e artes, desde o pré-escolar ao ensino secundário.

De acordo com alguns autores, (Chou, 2018; Körber et al., 2021; Žáček & Smolka, 2019), as atividades desenvolvidas com o robô Ozobot, apontam para a promoção das dimensões de Pensamento Computacional em crianças do ensino básico, aliás tema central geral desta investigação. A título de exemplo, e usando como recurso as atividades trabalhadas com o robô Ozobot, podemos falar da abstração quando um aluno representa o comportamento complexo num conjunto de blocos de programação, ou o raciocínio lógico face a uma dada situação, em que o aluno vai optar por determinadas opções em detrimento de outras “se eu usar este código, o Ozobot vai se comportar desta forma...”, mesmo na decomposição, quando um aluno programa uma dança, ou segue um caminho de vários, passos programa cada passo de forma separada ou a depuração do código no final do exercício, onde compara o comportamento esperado com os resultados obtidos e/ou fornece soluções alternativas mais eficientes.

3. METODOLOGIA

Robótica Educativa, um contributo para o desenvolvimento do pensamento computacional na escolaridade inicial

Os objetivos deste trabalho foram já delineados no parâmetro da Introdução, sendo reveladas, neste capítulo, as principais considerações de índole metodológica com vista a atingir esses mesmos objetivos, bem como oferecer uma descrição da sua planificação e implementação do estudo realizado. Assim, far-se-á uma apresentação do plano de atividades realizadas no âmbito da intervenção pedagógica e sua estruturação global, caracterizando-se os materiais pedagógicos desenvolvidos, bem como, indicando-se e justificando-se a utilização dos instrumentos de validação e métodos de recolha e tratamento de dados, para a investigação. Serão ainda abordados alguns aspetos referentes à preparação do trabalho de campo que permitiu a implementação do estudo de caso, realização, a sua análise e posterior conclusão.

3.1. Opção metodológica

Nesta secção, apresenta-se a opção metodológica do estudo realizado. Baseou-se numa metodologia de investigação qualitativa e numa abordagem interpretativa, recorrendo a um estudo de caso, de forma a compreender e conhecer as especificidades do desenvolvimento das habilidades do pensamento computacional em crianças do 1.º ciclo. No método qualitativo, o estudo caracteriza-se pelas estratégias próprias de investigação-ação. Denzin & Lincoln (2017), referem que uma pesquisa qualitativa consiste num conjunto de práticas interpretativas e materiais empíricos que tornam o mundo visível: – estudos de caso; introspeção; entrevistas; artefactos físicos; análise documental; registo de campo; registo audiovisual; inquéritos; entre outros; – que descrevem momentos significativos e determinadas rotinas na vida dos sujeitos. Por conseguinte, os investigadores utilizam uma ampla variedade de práticas interpretativas interligadas com o objetivo de compreender melhor um determinado tema.

Portanto, numa investigação pretende-se que o investigador se situe no paradigma interpretativo, em explorar e analisar as estruturas que geram o conteúdo específico e único na construção do conhecimento. As interpretações que o investigador cria com base nos relatos e interpretações expressados pelos próprios participantes sobre os fenómenos em estudo. Procurando as “respostas” para o tema/problema em questão.

Tendo em conta a problemática em estudo, os objetivos pretendidos, o “*timing*” de execução e o estudo de natureza qualitativa e descritiva, optámos por um desenho de estudo de caso. De acordo com Yin, (2015), um estudo de caso é uma investigação empírica que se foca num fenómeno contemporâneo num contexto real, permitindo uma generalização dos resultados. Também segundo o autor, existem condições relevantes que justificam a decisão da opção por um estudo de caso como estratégia, nomeadamente, a construção de uma questão de investigação, que possibilite responder às perguntas do género “como?” e “porquê?”, aqui acrescentaria o “Para quem?”.

Para Yin (2015), um estudo de caso “é uma investigação empírica que investiga um fenómeno contemporâneo dentro do seu contexto de vida real especialmente quando os limites entre o fenómeno e o contexto não estão claramente definidos”. Para este autor, um estudo de caso deve ser considerado quando existe a pretensão de satisfazer as condições contextuais, porque se acredita que são relevantes para o fenómeno em estudo.

Dooley, em 2002 citado em Meirinhos & Osório (2010), reforça que o estudo de caso tem a vantagem da “sua aplicabilidade a situações humanas, a contextos contemporâneos de vida real.”.

3.2. Descrição do estudo

O *design* do presente estudo reuniu o desenvolvimento de procedimentos de planeamento (por uma equipa formada pelo investigador, dois docentes do grupo 540 e docentes do 1.º ciclo) e investigação sobre um ambiente de aprendizagem inserido num contexto social específico, numa sala de aula com características únicas, procurando ser, simultaneamente, uma investigação no âmbito educativo e científico. Pretendeu-se desenvolver o projeto em duas etapas: 1 - Realização da experiência de ensino em contexto de sala de aula; 2 - Avaliação da experiência.

É a partir desta sequência que se torna importante uma análise de dados e uma reflexão, possibilitando reformulações, novas planificações, assim como, novas formas de intervenção. Esta investigação foi aplicada na disciplina de OCPR, para os alunos do 1.º CEB do agrupamento, durante o ano letivo 2021/2022. A disciplina de OCPR foi integrada no projeto curricular do agrupamento de escolas para entrar em funcionamento no ano letivo 2015-2016. O respetivo programa foi aprovado pelo conselho pedagógico, de 16 de julho de 2015. Assim, pretendeu-se realizar um conjunto de ações, de forma a criar um cenário de aprendizagem com uma visão de inovação e de práticas pedagógicas avançadas com a RE para o desenvolvimento do PC nas crianças.

Figura 15 - Visão global das diversas fases do estudo e instrumentos de validação



Fonte: Imagem trabalhada pelo investigador

A figura 15 evidencia o desenho global do estudo, onde se ilustra o esquema geral de execução e validação das atividades propostas em cada um dos momentos e a identificação dos principais instrumentos usados na validação.

O objetivo da fase de implementação é o de fornecer aos alunos um conjunto de atividades com conjuntos de RE que promova as dimensões de PC (abstração, decomposição, reconhecimento de padrões, algoritmos e depuração) através de conceitos das ciências da computação (sequências, variáveis, ciclos, condições e eventos). Por sua vez, o objetivo da fase de validação consiste em diversos momentos de recolha de dados (2 - Teste piloto, 3 – pré-teste, 4 - pós-teste e 5 – *Focus group*) por forma a dar resposta aos objetivos desta investigação.

Em termos de horário, de acordo com o documento Projeto de Desenvolvimento do Currículo do Agrupamento (PDCA), as aulas de OCPR decorrem em blocos de 50 minutos, em simultaneidade com apoio ao estudo, lecionado pela professora titular de turma; por conseguinte, a turma está dividida em turnos A/B. Enquanto um grupo de alunos está na aula de Programação e Robótica, o outro grupo está com a professora titular. Se esta divisão permite ao professor uma melhor organização e gestão da sala de aula e uma melhoria na escolha de estratégias e de práticas pedagógicas, por outro lado, em termos práticos, uma sessão de aula prolonga-se por duas semanas, o que poderá limitar o número de conteúdos a lecionar.

Integrando o horário letivo dos alunos, as aulas de OCPR estão distribuídas semanalmente de acordo com a tabela seguinte:

Tabela 1 - Horário das sessões de OCPR

	2 ^a	3 ^a	4 ^a	5 ^a	6
14:00/14:50	4A 3AP			3B	
15:00/15:50	4B 4AP			3A	
Turmas 3A,3B,4A e 4B – Escola EB2 Turmas 3AP e 4AP - Escola EB1					

Fonte: Tabela trabalhada pelo investigador

Na fase inicial deste trabalho, foi necessário, previamente, resolver questões de ordem logística e realizar alguns procedimentos operacionais e organizacionais, que passaram pela seleção das turmas, pelo contacto com a direção do agrupamento, pelo contacto com os professores titulares e professores que lecionam a OCPR, pela autorização e uso do teste de PC, pela obtenção de autorizações dos encarregados de educação, pela disponibilização dos espaços em diferentes momentos da investigação e pela obtenção dos diferentes equipamentos (Conjuntos robóticos, computadores, *tablets* e outros).

Uma vez garantidas as condições para o início do estudo, prosseguiu-se com a aplicação da intervenção educativa no âmbito da RE.

A tabela 2 ilustra as diferentes fases desta intervenção e a sua relação com os objetivos de investigação subjacentes a este trabalho.

Tabela 2 - Fases do estudo e objetivos de investigação que se pretendem ver respondidos

Fases	Objetivos	
Fase 1 Programa de intervenção - Aprendizagem dos conceitos básicos de ciências de computação, alinhados com os padrões ISTE e CSTA (<i>K12 – Computer Science Framework</i>).	OBG	Caracterizar o processo de desenvolvimento de atividades de RE com crianças do 1.º ciclo do ensino básico, que promovam dimensões do PC.
	OBE1	Descrever metodologias a implementar.
	OBE2	Caracterizar materiais pedagógicos que suportem as atividades.
Fase2 Aplicação do teste de PC desenvolvido por Marcos González (Teste piloto, pré-teste e pós-teste).	OBE3	Identificar os níveis de desenvolvimento do PC.

Fonte: Tabela trabalhada pelo investigador

A tabela anterior apresenta uma visão geral do estudo nas duas fases contempladas, com indicação dos diversos tópicos abordados. O objetivo da fase 1 é dotar os alunos dos conceitos básicos das ciências da computação através de uma intervenção pedagógica de 10 sessões, bem como os conceitos básicos de manuseamento dos conjuntos de robótica educativa. A fase 2 está relacionada com os momentos significativos da aplicação de instrumentos que possibilitem medir o nível de desenvolvimento do Pensamento Computacional.

3.3. Local do estudo

O estudo foi realizado num Agrupamento de Escolas, o qual é constituído por quatro estabelecimentos de ensino: EB1, EB2, EB3 e Escola Secundária.

A investigação incidiu sobre a Escola EB1 e a Escola EB2. O estudo recaiu em três turmas do 3.º ano (3A, 3B e 3AP) e três turmas do 4.º ano (4A, 4B e 4P) deste Agrupamento de escolas.

Para além da disciplina de OCPR, a instituição oferece ainda as atividades de enriquecimento curricular (AEC) para o 1.º ciclo do ensino básico que integram o Inglês, a Atividade Física e Desportiva e as Expressões, bem como a Componente de Apoio à Família (CAF), atividades de expressão física e desporto, atividades de expressão musical e movimento e atividades de expressão plástica e trabalhos manuais.

3.4. Participantes

No ano letivo 2021/2022 frequentavam, neste agrupamento de ensino, 2333 alunos distribuídos pelos diferentes níveis/ciclos de ensino. De acordo com as informações retiradas do Projeto de Desenvolvimento do Currículo do Agrupamento (PDCA), ao nível da Educação Pré-Escolar e 1.º ciclo, estão repartidos da seguinte forma:

Tabela 3 - Distribuição da população do Pré-escolar e 1.º Ciclo

Grau de Ensino	N.º de Alunos
Educação Pré-Escolar (Escola EB2)	48
1.º Ciclo (Escola EB2)	182
1.º Ciclo (Escola EB1)	86
Total	316

Fonte: Tabela trabalhada pelo investigador

Os participantes no estudo estão categorizados nas tabelas seguintes (3 e 4) e envolvem professores e alunos.

Tabela 4 - Distribuição de professores

Professor	N.º de Professores
Titular	6
Oferta (OCPR) - G540	2
Investigador	1

Fonte: Tabela trabalhada pelo investigador

Tabela 5 - Distribuição do número de alunos por turma

Turma	N.º de Alunos
3ª	24
3B	25
3AP	24
4ª	24
4B	23
4AP	19
TOTAL	139

Fonte: Tabela trabalhada pelo investigador

A iniciação à programação e robótica é entendida como mais uma ferramenta ao serviço, em articulação com as restantes áreas curriculares e não como mais uma área disciplinar. O trabalho desenvolvido foi, tanto quanto possível, articulado com o professor titular da turma, com a coadjuvância de docentes do

grupo disciplinar 540 – Eletrotécnia e com o investigador na negociação de uma proposta de desenho de investigação que permitiu este estudo. Os participantes no estudo foram um grupo de professores e o próprio investigador (tabela 4) e um grupo de alunos do 3.º e 4.º anos do AEFH (tabela 5), com idades compreendidas entre os 7 e os 10 anos.

Desde cedo, estava presente a preocupação de conseguir adotar uma metodologia de investigação definida como uma abordagem subjetiva subordinada às múltiplas interpretações dos alunos num contexto específico. Os sujeitos participantes foram escolhidos pelo investigador por questões de proximidade, acessibilidade e objetivo do presente estudo. A opção por este critério diz respeito à vivência e relações estabelecidas pelo investigador com os saberes pedagógicos, científicos e experimentais que alicerçam a formação e, por conseguinte, a prática docente, Tardif (2010). Foi, previamente, decidido que os sujeitos participantes seriam alunos do 1.º ciclo, tendo sido este critério definido pelo investigador de selecionar deliberadamente/intencionalmente os participantes, pois considera que estes elementos possuem as características que são típicas ou representativas da população. Podemos considerar que se trata de uma amostragem por conveniência que normalmente é adequada e frequentemente utilizada para germinação de ideias em pesquisas exploratórias, essencialmente, Mattar & Ramos (2021).

3.5. Materiais pedagógicos

Nesta secção, faz-se a descrição dos materiais utilizados nas atividades planeadas para o estudo realizado no âmbito desta investigação. Inicia-se com uma visão geral da planificação estabelecida com as professoras cooperantes que lecionam a OCPR, que, por sua vez, articularam com as professoras titulares de turma. Procura-se assim, descrever os recursos envolvidos e respetivas atividades sugeridas pelo investigador para a realização do trabalho. As atividades dinamizadas são diferentes conforme os conteúdos a abordar e o interesse dos demais intervenientes. Procurou-se que o plano definido e os recursos envolvidos fossem aplicáveis numa perspetiva futura e não apenas por si só um resultado tangível deste trabalho. Um aspeto importante a salientar nesta secção é a tomada de opções de implementação em concreto, nomeadamente nos cuidados necessários para a implementação do estudo.

Tabela 6 - Fase de implementação do estudo, resumo das sessões

Sessão	Kit Robótico	Atividade	Conteúdos
1	LEGO WeDo 2.0	Apresentação da LEGO WeDo 2.0	Conceitos gerais, exploração das peças legos e aplicação de programação em blocos.
2	LEGO WeDo 2.0	Milo, o Robô Explorador da Ciência	Projeto introdutório dividido em 4 partes para trabalhar com as funções básicas do WeDo 2.0.
3	Placa Micro:Bit	Apresentação da placa Micro:Bit	Conceitos gerais, exploração do ambiente de programação makecode, com pequenos projetos exemplo.
4	Placa Micro:Bit	Exploração de variáveis com Micro:Bit	Trabalhar com dados/variáveis, o que são variáveis, para que servem e exercícios <i>unplugged</i> e programados.
5	Placa Micro:Bit	Testar condições com a placa Micro:Bit	Trabalhar com condições/condicionais, o que são e para que servem e exercícios <i>unplugged</i> e programados.
6	Placa Micro:Bit	O que são iterações?	Trabalhar com ciclos/iterações, o que são e para que servem e exercícios <i>unplugged</i> e programados
7	Placa Micro:Bit	Mini-Projeto "Hot Potato"	Mini-projeto que incorpora programação com variáveis, ciclos, números aleatórios e som.
8	Placa Micro:Bit	Mini-Projeto "Buzzer Game"	Mini-projeto que incorpora fios elétricos e trabalha com os pinos de entrada do Micro:Bit, variáveis, ciclos, condições e som.
9	Robô Ozobot	Apresentação do Robô Ozobot, programação com cores	Conceitos gerais, exploração da programação por canetas de cor.
10	Robô Ozobot	Introdução ao Ozobot Blockly Conceitos Básicos	Programação com blocos, exploração das categorias: ciclos, som, tempo, movimento, e efeito de luzes.

Nota: Cada sessão é realizada numa semana ao turno A e na semana seguinte ao turno B

Fonte: Tabela trabalhada pelo investigador

A tabela 6 apresenta uma visão global deste estudo, um trabalho realizado, que contou com o apoio das colegas que lecionam a disciplina de OCPR, em parceria com os professores titulares da turma. Nas várias sessões, sempre que foi introduzido um novo conjunto robótico (LEGO WeDo 2.0, Micro:Bit e Ozobot) foram utilizadas abordagens mais interativas e menos expositivas, introduzindo os conceitos à

medida das necessidades, de acordo com a progressão dos alunos e do seu nível cognitivo, tendo por base a abordagem UMC referenciada na secção 2.2.

O objetivo é o de fornecer aos alunos um conjunto de saberes iniciais de conceitos básicos de robótica e ciências da computação, tendo uma duração prevista de 10 sessões de 45 minutos; no entanto, realça-se que algumas aulas de OCPR foram destinadas a preparar outras atividades, como é o exemplo da peça de teatro sobre a Segurança na Internet e as questões do *Cyberbullying*. Importa realçar que as atividades estão constantemente ligadas, foram trabalhadas pelos alunos do 3.º e 4.º anos de escolaridade, sob a forma de um teatro colaborativo, no âmbito da Oferta de Escola de Programação e Robótica, descritas na secção (3.5.2) em atividades complementares.

As atividades propostas seguem as linhas orientadoras do Ministério da Educação que através do perfil dos alunos à saída da escolaridade obrigatória, do decreto-Lei n.º 55/2018, de 6 de julho e o despacho n.º 6944-A/2018, de 19 de julho, que enquadra as TIC como áreas de integração curricular transversal no 1.º CEB. Estes documentos visam a necessidade de desenvolver competências que permitam aos alunos acompanhar o desenvolvimento tecnológico emergente dos nossos dias. Para além destas linhas orientadoras foram considerados outros documentos como as metodologias aplicadas ao projeto PROBÓTICA - Programação e robótica no ensino básico (Pedro et al., 2017) e em termos de conteúdos curriculares relacionados com as ciências da computação, em determinados conceitos. Estes deverão ser vistos como uma contextualização transversal de todo o processo educativo, neste sentido, foram retirados das propostas educativas “K-12 *Computer Science Framework*” (<https://k12cs.org/wp-content/uploads/2016/09/K%E2%80%9312-Computer-Science-Framework.pdf>), “K-12 *Computer Science Standards*” (<https://drive.google.com/file/d/1-dPTAI1yk2HYPKUWZ6DqaM6aVUDa9iby/view>) e “*ISTE Computational Thinking Competencies*” (<https://www.iste.org/standards/iste-standards-for-computational-thinking>) e/ou adaptados, de acordo com a nossa realidade educativa.

O K-12 *Computer Science Framework*, foi desenvolvido por especialistas em diferentes áreas da Educação, Computação com o envolvimento da comunidade a nível mundial e contém diretrizes conceptuais para o ensino das ciências da computação servindo como um guia para o desenvolvimento de currículos personalizados. Este documento orientador promove uma visão na qual todos os alunos que estejam envolvidos de uma forma crítica nos assuntos relacionados com as ciências da computação, sejam capazes de abordar problemas de formas inovadoras e criem artefactos computacionais numa abordagem mais prática, pessoal e social (CSTA, 2016). Neste sentido, a contextualização das ciências da computação, tal como acontece em outras áreas educativas, deverá ser trabalhada de forma que os

alunos possam compreender situações do quotidiano e que as mesmas sejam relevantes para o presente e para o futuro de cada um.

O documento organiza o conhecimento das ciências da computação em **conceitos fundamentais** que representam as áreas nucleares (5) das ciências da computação e **práticas fundamentais** que representam as ações (7) que os alunos praticam de forma envolvente, com o objetivo de aquisição de conceitos de forma significativa (tabela 7). Cada conceito fundamental contém uma série de subconceitos que estão relacionados com uma ou mais práticas fundamentais.

Tabela 7 - Estrutura do Framework das Ciências da Computação K-12

Conceitos Fundamentais		Práticas Fundamentais	
1	Sistemas de Computação	1	Promover uma cultura de computação inclusiva
2	Redes e Internet	2	Colaborar à volta da computação
3	Dados e Análise	3	Reconhecer e definir problemas computacionais
4	Algoritmos e Programação	4	Desenvolver e usar abstrações
5	Impactos da Computação	5	Criar artefactos computacionais
		6	Testar e refinar artefactos computacionais
		7	Comunicar sobre a computação

Fonte: Tabela trabalhada pelo investigador, adaptado de (CSTA, 2016)

Obviamente que, para este estudo no programa de intervenção, foram considerados apenas os conceitos fundamentais relacionados com **Algoritmos e Programação**, envolvendo as práticas fundamentais (3, 4, 5 e 6) que circunscrevem o **pensamento computacional** e (1,2 e 7) como independentes que complementam o pensamento computacional, abordados de uma forma transversal (CSTA, 2016, p.67). A construção de materiais pedagógicos teve em consideração o K-12 *Computer Science Standards*, que é um documento baseado K-12 *Computer Science Framework*, de uma forma prática e resumida que fornece uma expectativa detalhada e mensurável do desempenho dos alunos. Nesse documento, encontra-se uma divisão por níveis de ensino, conforme (tabela 8).

Tabela 8 - Níveis de ensino de acordo com o CSTA

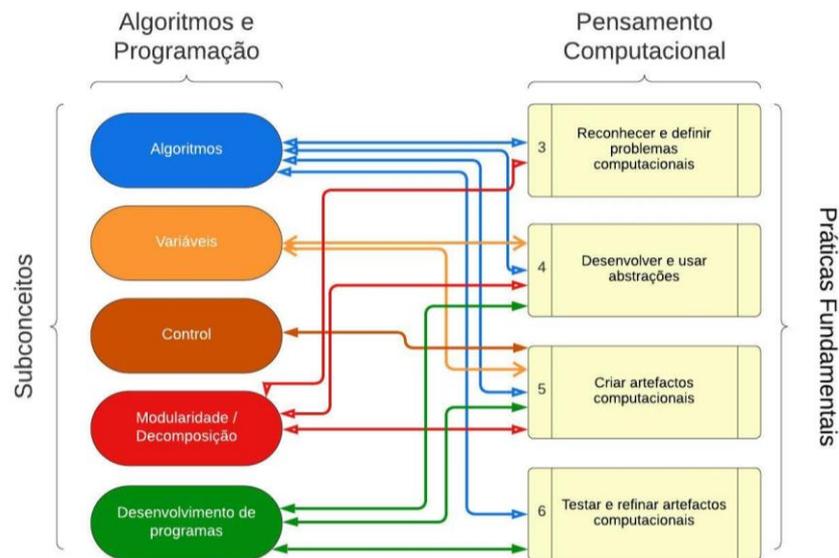
Nível 1	Nível 2	Nível 3
1A 5 a 7 anos	11 a 14 anos	3A 14 a 16 anos
1B 8 a 11 anos		3B 16 a 18 anos

Fonte: Tabela trabalhada pelo investigador, adaptado de (CSTA, 2016)

Como a investigação incidiu sobre os 3.º e 4.º anos do ensino básico, o foco será centrado nos conteúdos para o nível 1B, para idades compreendidas entre os 8 e os 11 anos. Assim, no contexto deste trabalho, é explorada a relação entre os conceitos fundamentais de **Algoritmos e Programação** (e subconceitos)

e as práticas fundamentais do **Pensamento Computacional** proposto pelo *K-12 Computer Science Standards*, espelhada na grelha de progressão (<https://csteachers.org/documents/en-us/d227e2a4-ce35-434e-a20b-24355d11b015/1/>) revista em 2017 (CSTA, 2017) apresentada na figura 16. Tentando relacionar os conceitos fundamentais com as práticas fundamentais, nomeadamente algoritmos e programação (e subconceitos) e pensamento computacional, foi criada a seguinte ilustração com base na grelha de progressão (CSTA, 2017).

Figura 16 - Conceitos e práticas fundamentais CSTA



Fonte: Elaborado pelo investigador com base na tabela de progressão em (CSTA, 2017)

Tabela 10 - Objetivos de aprendizagem por subconceito CSTA

Conceito Fundamental	Subconceito	Objetivos a atingir
Algoritmos e Programação	Algoritmos	1B-AP-08 - Comparar e aperfeiçoar dentre vários algoritmos para a mesma tarefa e determinar qual é o mais adequado. (P6.3, P3.3)
	Variáveis	1B-AP-09 - Criar programas que usam variáveis para armazenar e modificar dados. (P5.2)
	Controlo	1B-AP-10 - Criar programas que incluam sequências, eventos, ciclos e estruturas condicionais. (P5.2)
	Modularidade	1B-AP-11 - Decompor (partir) problemas em problemas de menor dimensão, de forma a facilitar o processo de desenvolvimento do programa. (P3.2)
		1B-AP-12 - Modificar, misturar ou incorporar partes de um programa já existente no seu próprio trabalho, para desenvolver algo novo ou adicionar recursos mais avançados. (P5.3)
	Desenvolvimento de programas	1B-AP-13 - Usar um processo iterativo para planear o desenvolvimento de um programa, incluindo as perspectivas dos outros e considerando as preferências de utilizador. (P1.1, P5.1)
		1B-AP-14 - Observar os direitos de propriedade intelectual e atribuir o devido reconhecimento ao criar ou misturar programas. (P7.3)
1B-AP-15 - Testar e depurar (identificar e corrigir) erros de um programa ou algoritmo para garantir que ele seja executado conforme planeado. (P6.1, P6.2)		
		1B-AP-16 - Assumir papéis diferenciados, com orientação do professor, ao colaborar com colegas durante o desenvolvimento do projeto, implementação e etapas de revisão do desenvolvimento do programa. (P2.2)
		1B-AP-17 - Descrever as escolhas efetuadas durante o desenvolvimento do programa usando comentários no código, apresentações e demonstrações. (P7.2)

P3 - Reconhecer e definir problemas computacionais | P4 - Desenvolver e usar abstrações

P5 - Criar artefactos computacionais | P6 - Testar e refinar artefactos computacionais

Fonte: Elaborado pelo investigador com base na tabela de progressão em (CSTA, 2017)

A tabela 10 ilustra uma tradução da grelha de progressão, nomeadamente dos conceitos, subconceitos e objetivos de aprendizagem a atingir em cada subconceito, para alunos com idades compreendidas entre os 8 e os 11 anos (Nível 1B).

Os materiais usados nas sessões estão planejados para despertar o interesse dos alunos pela área da robótica e da programação, possibilitando a aprendizagem de conceitos básicos das ciências da computação, tais como: **sequências, dados/variáveis, operadores, condições/estruturas de decisão, ciclos/iterações/repetições e eventos**. Estes conceitos serão trabalhados pelos alunos recorrendo a conjuntos robóticos e, em cada sessão, os conteúdos serão introduzidos de forma gradual e sequencial, e em algumas situações poderá ser necessário mais do que uma sessão para que seja feita uma consolidação das aprendizagens.

Figura 17 - Conjunto de atividades aplicadas durante o programa de intervenção

Sessão	Kit Robótico	Atividade	Sequências	Eventos	Ciclos	Variáveis	Operadores	Condições
1	LEGO WeDo 2.0	Apresentação da LEGO WeDo 2.0	x	x			x	
2	LEGO WeDo 2.0	Milo, o Robô Explorador da Ciência	x	x	x		x	
3	Placa Micro:Bit	Apresentação da placa Micro:Bit	x	x	x			
4	Placa Micro:Bit	Exploração de variáveis com Micro:Bit	x	x	x	x	x	
5	Placa Micro:Bit	Testar condições com a placa Micro:Bit	x	x	x			
6	Placa Micro:Bit	O que são iterações?	x	x	x	x	x	
7	Placa Micro:Bit	Mini-Projeto "Hot Potato"	x	x	x	x	x	x
8	Placa Micro:Bit	Mini-Projeto "Buzzer Game"	x	x	x	x	x	x
9	Robô Ozobot	Apresentação do Robô Ozobot, programação com cores	x	x				
10	Robô Ozobot	Introdução ao Ozobot Blockly - Conceitos Básicos	x	x	x	x	x	x

Fonte: Elaborado pelo investigador e professores cooperantes

Os exercícios apresentados na tabela anterior, na opinião do investigador em consonância com as professoras cooperantes, estão organizados de forma a proporcionar uma aprendizagem de carácter prático, com o intuito de criar um ambiente educativo menos formal e, conseqüentemente, mais

significativo para os alunos com base nos pressupostos da estratégia de desenvolvimento curricular UMC apresentada na secção 2.2.

As sessões de trabalho tiveram, permanentemente, uma parte introdutória explicativa por parte do professor, normalmente exemplificando o modelo ou programa existente, e, a partir desse passo, os alunos exploraram, modificaram e criaram.

3.5.1. Materiais desenvolvidos para o estudo

Nesta secção, em conformidade com os pressupostos apresentados anteriormente, faz-se a descrição dos materiais desenvolvidos para o estudo, explicitando as atividades aplicadas em cada uma das sessões. Procura-se descrever para cada atividade da **Fase 1** (tabela 2 e tabela 6) que está, diretamente, relacionada com a abordagem dos objetivos deste trabalho, nomeadamente o **OBE2** (Caracterizar materiais pedagógicos que suportem as atividades). Assim, esta fase tem como foco possibilitar aos alunos a aquisição de um conjunto de competências diretamente relacionadas com a construção, exploração e programação de robôs, de forma a promover o desenvolvimento do pensamento computacional.

Por outro lado, é expectável que o desempenho dos alunos na realização das diversas atividades deverá permitir ao investigador retirar dados com vista ao esclarecimento do objetivo **OBE1** (Descrever metodologias a implementar). No final da Fase 1, os alunos realizaram um teste para aferição das habilidades do pensamento computacional (Fase 2), tendo como alvo a resposta ao **OBE3** (Identificar os níveis de desenvolvimento do PC).

Sessão 1 - Apresentação do conjunto LEGO WeDo 2.0 – Apresentação do kit aos alunos (1 caixa por grupo de dois alunos); conhecer o kit robótico, apresentação dos componentes (*hub*, motores, sensores e peças de construção) e proposta de atividade conjunta de exploração passo a passo (**ANEXO E1**).

O conjunto LEGO Education WeDo 2.0, relaciona as possibilidades da utilização das peças LEGO com os parâmetros curriculares das ciências e as competências científicas presentes no PISA (*Programme for International Student Assessment*). Os projetos que constam no curriculum do *WeDo 2.0* têm como objetivo trabalhar algumas competências científicas e tecnológicas, nomeadamente: fazer perguntas e resolver problemas; desenvolver (modelar) e usar modelos; projetar protótipos; realizar investigações; analisar e interpretar dados; usar a Matemática e o pensamento computacional; usar evidência científica;

obter, avaliar, argumentar e comunicar; identificar questões científicas; explicar fenômenos cientificamente (Lego, 2017).

Nesta sessão, para além da atividade em conjunto, foram dadas instruções essenciais para manipulação do *Smarthub*, nomeadamente: identificar e etiquetar os vários separadores da caixa; identificar o microcontrolador, o motor e os sensores com um número (facilita a distribuição de cada conjunto pelo grupo de alunos), alterar o nome no *software*, de forma a facilitar o processo de emparelhamento *Bluetooth* através do nome (por exemplo: wedo_#1);

Como sugestão do trabalho em grupo, cada aluno pode ter uma função específica dentro do grupo, por exemplo: construtor, aquele que identifica e encaixa as peças; programador, aquele que trata de definir as sequências do programa; relator, o que tira fotografias, faz vídeos e relata a investigação; apresentador, aquele que explica o projeto; capitão, o representante do grupo. Pode fazer sentido, também, alternar as funções no seio do grupo, de forma a deixar que cada aluno experimente todos os momentos do projeto e, assim, tenha oportunidade de desenvolver uma série de habilidades. Ao longo da sessão, as professoras cooperantes podem recorrer a várias questões para verificar se os alunos estão a assimilar e a proceder de forma correta, promovendo a discussão em grupo sobre várias questões colocadas, já que o objetivo desta sessão passa por explorar as diferentes peças do kit, construir um pequeno artefacto, ligar o *Hub* à aplicação de *software*, explorar o *software* e criar/experimentar a codificação por blocos de forma a observar o efeito produzido. Estes pressupostos serão a base para a atividade da sessão 2, uma vez que o progresso cognitivo é baseado nos conceitos previamente adquiridos.

Sessão 2 - Milo, o Robô Explorador da Ciência – Nesta sessão é dada continuidade à exploração do conjunto LEGO WeDo 2.0, mas com a finalidade de desenvolver um projeto em que os alunos descubram diferentes formas utilizadas pelos cientistas e engenheiros no uso de sondas para explorar lugares onde os humanos não conseguem ir. Este projeto introdutório, proposto no Teacher Guide LEGO WeDo 2.0 (Lego, 2017), é composto por 4 partes: A – Milo, o Robô explorador da ciência; B – Sensor de movimento; C – Sensor de inclinação; D – colaboração. Uma proposta de atividade conjunta de exploração passo a passo encontra-se em anexo a este trabalho (**ANEXO E2**), no entanto, um bom guia para os professores poderá ser encontrado nestes 4 planos de aulas (<https://education.lego.com/pt-br/lessons/wedo-2-science>) do currículo da LEGO® Education WeDo 2.0. A sessão inicia-se com um pequeno vídeo introdutório sobre a temática, nomeadamente como os engenheiros e cientistas partem à descoberta de lugares inexplorados e a necessidade de desenvolver vários mecanismos (satélites, sondas, drones, naves espaciais e robôs), finalizando com um pequeno debate sobre o assunto.

Após a tarefa anterior, os alunos exploram quatro versões do robô Milo, sempre com pequenos vídeos de ajuda e instruções passo a passo para a construção/alteração do robô. Depois de estar construído terão de o programar (por exemplo, ligar o motor com potência de 8, deslocar numa dada direção por 2s e depois parar). Durante a fase de programação, os alunos têm a oportunidade de irem à Biblioteca de design (aplicação do LEGO WeDo 2.0) e obter inspiração com outras sequências de programação e verificarem diferentes funcionalidades.

De acordo com as diretrizes dos Projetos Curriculares LEGO® Education WeDo 2.0 (Lego, 2016) as atividades realizadas estão associadas a questões do quotidiano dos alunos, desafiando-os a resolver problemas realistas enquanto utilizam peças LEGO, sensores, atuadores e programação.

O LEGO WeDo 2.0 é baseado no mais recente currículo de ciências e foi criado para melhorar a curiosidade dos alunos e as habilidades na área das ciências da computação, abrangendo as áreas das ciências, engenharia, tecnologia e programação. O conjunto robótico, reforça a compreensão dos alunos em fenómenos em diferentes áreas da ciência (física, biologia, aeroespacial), na resolução de problemas, na modelagem e prototipagem, na análise e interpretação de fenómenos/dados, no desenvolvimento das habilidades do pensamento computacional, o espírito crítico e o trabalho em grupo.

De acordo com, Chalmers (2018), os conjuntos da LEGO WeDo 2.0 são apropriados para o desenvolvimento de conceitos básicos de programação e do pensamento computacional. Todos os professores participantes no seu estudo, relatam que as atividades que envolveram estes robôs, ofereceram oportunidades únicas para o desenvolvimento das competências do PC, nomeadamente aquelas que promovem a resolução de problemas e o trabalho em grupo.

Sessão 3 - Apresentação da placa Micro:Bit – Todos os exercícios que envolvem a placa Micro:Bit, são baseados no curso patrocinado e publicado pela Microsoft designado - *Introduction to Computer Science course*. O conteúdo curricular é da autoria de Douglas Kiang e Mary Kiang (Kiang & Kiang, 2019) e estão de acordo com os padrões de Ciências da Computação CSTA K-12 (CSTA, 2017), revistos em 2017. É um currículo baseado em projetos que tem no seu *core* a filosofia *maker*, assenta na ideia de que, ao conceber artefactos físicos, os alunos criam um contexto para aprender os conceitos de básicos das ciências da computação.

Nesta sessão, os alunos têm o primeiro contacto com a plataforma MakeCode (<https://makecode.microbit.org/#editor>) que dá suporte ao simulador. Esta plataforma apresenta de forma visível as alterações sempre um bloco de programação for inserido ou removido, o que possibilita

obter resultados como se tratasse da placa física, permitindo a criação de projetos divertidos com resultados imediatos. Depois da explicação da plataforma por parte das professoras, os alunos são convidados a utilizar a placa física e a transferir os seus projetos, comparando os resultados físicos com os simulados. Os objetivos desta sessão passam pela familiarização do ambiente de programação da plataforma MakeCode, ser capaz de efetuar o *download* do programa para a placa física Micro:Bit e explorar os botões e os sensores da placa **(ANEXO E3)**.

Sessão 4 - Exploração de variáveis com Micro:Bit – Nesta sessão, dando continuidade à aprendizagem das ciências da computação por intermédio da placa Micro:Bit, o foco centra-se no conceito de dados/variáveis **(ANEXO E4)**. O objetivo desta sessão foi o de promover o conceito de variáveis, nomeadamente, os alunos devem: ter a noção do que são variáveis e porquê e quando devem ser usadas num programa; aprender como criar uma variável, definir um valor inicial numa variável e alterar esse valor no decorrer do programa; aprender a criar nomes de variáveis com significado e com elegíveis; perceber que as variáveis contêm apenas um valor num dado momento; ter consciência que ao atualizar/alterar o valor de uma variável, o novo valor substitui o anterior; reconhecer e usar os blocos de operadores matemáticos básicos como adicionar, subtrair, multiplicar e dividir juntamente com às variáveis; reconhecer a importância do uso de variáveis e aplicar esse conhecimento no desenvolvimento de programas.

Todas as fichas de trabalho têm uma introdução sobre os objetivos de aprendizagem, sendo espectável um esclarecimento por parte das professoras aos alunos. Em alguns casos, poderá existir uma atividade específica que pode contribuir para a apresentar um determinado assunto aos alunos de uma forma divertida. A sessão inicia com uma atividade sem computadores (*unplugged*) que não requer programação ou a placa Micro:Bit. O uso de atividades desligadas ou sem recurso a computadores é uma oportunidade de os alunos aprenderem conceitos das ciências da computação de uma forma tangível e com o próprio movimento do corpo, neste caso específico jogar o jogo pedra, papel, tesoura e registar os resultados numa folha de papel. Este procedimento pode ser benéfico, pois muitos dos conceitos das ciências da computação são abstratos e poderá ser uma forma diferente de compreensão (Brennan & Resnick, 2012, p.6).

Depois dessa fase introdutória os alunos trabalham no simulador e na placa Micro:Bit, seguindo orientações passo a passo, com o objetivo de aplicarem os conceitos adquiridos/compreendidos no exercício *unplugged* anterior.

No final da atividade, os alunos terão um programa escrito que podem reutilizar num projeto próprio ou concretizar os desafios lançados no final de cada ficha. Para aqueles alunos que terminarem mais cedo as atividades, poderão ajudar os colegas que estão mais atrasados ou conceberem algo novo pois, sentem-se confiantes para aceitar novos desafios, de acordo com a progressão dos alunos e do seu nível cognitivo, tendo por base a progressão UMC descrita na secção 2.2.

Sessão 5 - Testar condições com a placa Micro:Bit – Nesta sessão, o foco são as condições simples, explorando os blocos lógicos “*If...then*” e “*If...then...else*”, recorrendo à placa Micro:Bit (**ANEXO E5**). Foram trabalhadas pelos alunos as estruturas condicionais. No final, os alunos devem: compreender o que são as estruturas condicionais, porque são importantes e em que situações devem ser utilizadas num programa; aprender a utilizar os blocos de programação “*If...then*” e “*If...then...else*”; demonstrar que compreenderam os conceitos e em que situações as estruturas condicionais diferentes produzem resultados específicos; construir um pequeno jogo (cara ou coroa) com a placa Micro:Bit recorrendo às estruturas condicionais.

A ficha de trabalho inicia com uma discussão em grupo sobre a utilização das condições no dia-a-dia, por exemplo quando os pais dos alunos dizem “**Se não limpares o teu quarto, não podes jogar no teu computador**”. Isto é uma frase que inclui uma estrutura de condição “*If...then*”, se a condição for verdadeira (alunos limparem o quarto), será executada uma ação (podem jogar). Depois a interiorizarem como funcionam as condições realizam uma atividade *offline/unplugged*, para consolidação dos conceitos. Nesta atividade, alguns alunos de forma aleatória serão alinhados numa extremidade da sala de aula com o objetivo de alcançar o outro lado da sala de aula. Pretende-se “chamar” os alunos mediante instruções usando diferentes condições. Os alunos visados avançarão ou não, dependendo da declaração condicional específica.

Na parte do simulador e na placa Micro:Bit, com orientações passo a passo, os alunos resolvem exercícios sobre os conceitos adquiridos. No final da atividade, os alunos construíram um pequeno jogo cara ou coroa sob a orientação das professoras cooperantes, neste exercício os alunos recorreram a conceitos já apreendidos em sessões anteriores e desta sessão. Com o jogo finalizado foi proposto aos alunos um acréscimo de dificuldade, nomeadamente adicionar código necessário para apresentar uma mensagem do nome do jogador que tem a pontuação mais alta.

Sessão 6 - O que são iterações? – Nesta sessão, o objetivo é abordar as estruturas de repetição/ iteração (*loops*) simples, explorando os blocos lógicos “*repeat*”, “*while*” e “*for*”, recorrendo à placa Micro:Bit (**ANEXO E6**). Foram trabalhadas pelos alunos as estruturas de repetição, no final os alunos

devem: entender o conceito de iteração em programação; compreender o significado de ciclo como uma forma de iteração; aprender como e quando usar os blocos cíclicos “*repeat*”, “*while*” e “*for*”; aplicar o conhecimento e as habilidades anteriores num programa criado pelos alunos com recursos as estruturas de repetição.

A ficha de trabalho inicia-se com uma discussão em grupo sobre a utilização das estruturas de repetição no dia-a-dia, por exemplo, quando lavamos o cabelo “**Que algoritmo podemos escrever para lavar o cabelo?**”. Após breve discussão, os alunos devem chegar a conclusão de que repetem determinadas instruções até lavar o cabelo, por exemplo (1 - cabelo molhado, 2- aplique o shampoo no cabelo molhado, 3 - esfregue o shampoo no cabelo e 4 – Enxague o shampoo do cabelo). Outros exemplos podem ser demonstrados, introduzido o conceito de que o ciclo repete o código até que uma determinada condição seja verificada ou a de um ciclo infinito.

Depois de interiorizarem como funciona o conceito de *loop*, realizam uma atividade *offline/unplugged*, para consolidação dos conceitos. Nesta atividade, em grupo de dois alunos, com uma cadeira ao centro da sala, um dos alunos indica as instruções para que o colega se desloque à volta da cadeira, desenhando um quadrado, um terceiro aluno escreve no quadro as instruções proferidas pelos colegas. Os alunos vão identificar que existem instruções que se repetem (reconhecimento de padrões) e que as estruturas de repetição alcançam um número menor de linhas de código para resolução do mesmo problema.

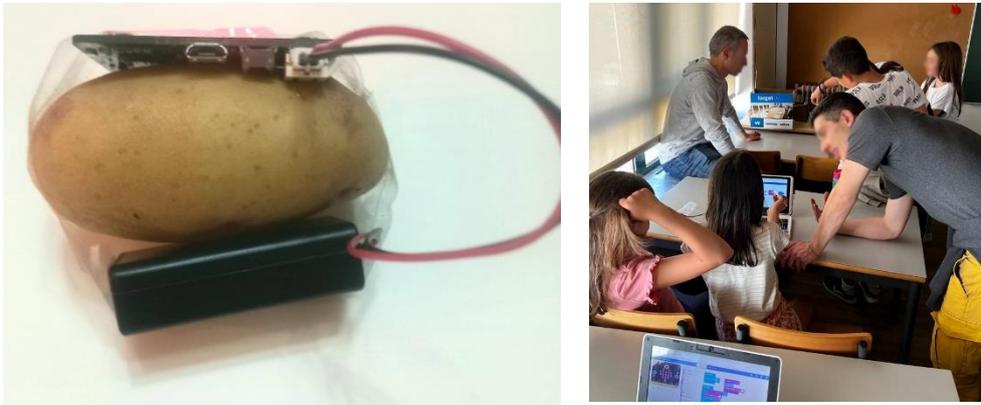
Na parte do simulador e na placa Micro:Bit, com orientações passo a passo, os alunos resolvem exercícios sobre os conceitos adquiridos através da abordagem UMC. No final da atividade, os alunos constroem um simples jogo, usando um pequeno ponto luminoso (*sprite*), que desenha um quadrado na matriz de leds da placa Micro:Bit. Como sugestão é proposto aos alunos um acréscimo de dificuldade, nomeadamente a alteração do código de forma a usar ciclos encadeados.

Sessão 7 - Mini-Projeto “Hot Potato”– Nesta sessão é proposta uma atividade de consolidação dos conhecimentos adquiridos nas sessões anteriores, sobre ciências da computação (algoritmos, variáveis, instruções de condicionantes e instruções de repetição), designada por jogo da batata quente (**ANEXO E7**). No início da aula, o professor deve fazer o ponto de situação e uma pequena revisão de conteúdos, se necessário. Deve explicitar aos alunos como funciona o jogo tradicional da batata quente, em que os jogadores se colocam em roda (de pé ou sentados) para ver quem consegue passar ao colega, um objeto o mais rapidamente possível, ao som de uma lengalenga sobre a “batata quente”. Quem ficar com o objeto na mão, no fim da lengalenga, sai. O jogo termina quando todos forem eliminados, e restar apenas o vencedor.

Após explicitação do algoritmo, os alunos, através do *software* da placa Micro:Bit, programam um conjunto de instruções passo a passo e, através da abordagem UMC, com o objetivo de reproduzir o jogo em computador.

Depois de testar o programa, os alunos foram convidados a colocar adereços e a alimentar a placa Micro:Bit, como aparece apresentado na imagem seguinte:

Figura 18 - Jogo "Hot Potato" com a placa Micro:Bit



Fonte: Imagem trabalhada pelo investigador, gentilmente cedida pelas professoras cooperantes

De forma a melhorar o engenho, os alunos foram convidados a pensar em sugestões de melhorias ao jogo, tais como: alterar o tempo de jogo até que a batata “expluda”, inserindo valores de forma aleatória; alterar o tipo de sons emitidos; alterar as animações, e outras ideias sugeridas pelos alunos.

Em Programação e Robótica brincamos, aprendemos, programamos, fazemos engenhocas.... e brincamos e aprendemos e programamos!.... Quem se atreve a jogar com a nossa "Batata Quente"?

Sessão 8 - Mini-Projeto “Buzzer Game”– Dando continuidade à consolidação dos conceitos básicos das ciências da computação através da placa Micro:Bit, é considerada uma proposta de aprendizagem por desenvolvimento de projeto tendo por base a estratégia UMC. Foi proposto aos alunos o desenvolvimento de um jogo tradicional intitulado “Jogo do arame” (**ANEXO E8**), que consiste em passar um aro de um material condutor por um fio curvilíneo do mesmo material sem lhe tocar. Face ao tipo de trabalho, é necessário previamente uma explicitação de eletricidade e eletrónica por parte das professoras cooperantes. Numa primeira fase, os alunos apenas necessitam da placa Micro:Bit, de um fio condutor, de um aro feito de fio condutor, plasticina para suporte do fio e pontas de crocodilo para fazer as ligações à placa Micro:Bit.

Depois de construírem o suporte para o jogo é necessário proceder à programação. Inicialmente uma codificação simples que indique visualmente na matriz de leds um “X” se tocar no fio, caso contrário,

apresentar o ícone de um *smile*. Também deverá ser adicionada a função se carregar no “botão A” limpa a matriz de leds.

Neste projeto, podem ser adicionadas novas funcionalidades e desafios, sempre que os alunos consigam resolver o passo anterior, por exemplo, pode ser sugerido que alterem a codificação para emitir um som sempre que tocar no fio, ou que acrescentem um componente eletrónico como um led que irá acender sempre que se verifique o toque no fio. Ou até criar uma variável que indique o número de toques.

Trabalhar em pequenos projetos com crianças e jovens, através de artefactos robóticos/eletrónicos só terá intencionalidade pedagógica e impacto significativo na aprendizagem, quando integrar de forma simples e transparente tecnologia com conteúdos e contextos, procurando mobilizar a tecnologia em contextos e problemas/desafios que possam ser explorados e trabalhados pelos alunos em abordagens transdisciplinares, desenvolvendo competências de comunicação, de planificação, de pensamento abstrato e de resolução de problemas.

Sessão 9 - Apresentação do Robô Ozobot, programação com cores – Será apresentado o robô aos alunos, com explicitação dos principais componentes de *hardware* e do seu funcionamento. Todos os exercícios que envolveram o robô Ozobot são baseados nas aulas disponibilizadas na plataforma Ozobot Classroom contendo uma planificação cuidada, um vídeo explicativo e instrutivo, adequado ao aluno, possibilitando o uso em sala de aula ou remotamente. Todas essas aulas foram testadas e validadas na prática, estando alinhadas com os objetivos e conceitos de aprendizagem definidos por diversos padrões como a CSTA e o ISTE, por forma a garantir que os alunos estejam a adquirir competências e aptidões do século XXI.

Esta sessão destina-se a explorar a programação por código de cores sem o recurso a computadores ou *tablets*, em que o robô Ozobot segue caminhos desenhados com cores que controlam a velocidade, direção, arranque, paragem e movimentos especiais. O objetivo da aula está direcionado para os alunos explorarem os códigos de cores orientados pela ficha de trabalho (**ANEXO E9**). São propostas pequenas atividades de desenho com marcadores que serão interpretados pelo sensor de cor do robô Ozobot. Os alunos iniciam com os comandos básicos sobre como usar o Ozobot: ligar/desligar, calibrar, desenhar linhas e programar com códigos de cores simples de acordo com a tabela de cores (<https://ozobot.com/create/color-codes/>), avançam para codificar diferentes tipos de velocidades, evoluindo para mais funcionalidades como movimentos especiais (tornado, ziguezague, para trás e girar) e terminam com atividades relacionadas com mudanças de direção e “saltos”, nomeadamente (ir em

frente, virar à esquerda, virar à direita, salto de linha para a frente, salto de linha para a esquerda, salto de linha para a direita, mudar de direção e chegar ao fim e voltar para trás).

Estes exercícios com programação por cores com o robô Ozobot são bastante motivadoras e aliciantes principalmente em crianças, tendo-se observado o trabalho colaborativo e a comunicação oral, na medida em que todas as atividades propostas às crianças trabalham em equipas, de forma colaborativa, com um objetivo comum o que faz com que as mesmas tenham de discutir em grupo e comunicar ideias.

Permitiu aos alunos que incluíssem o erro como parte integrante do processo de aprendizagem, pois é sempre possível corrigir o erro quando estamos a trabalhar com os robôs e temos facilmente a perceção do mesmo, ou seja, trata-se da conceção do erro como um processo pedagógico de aprendizagem e não de penalização por não ter sido capaz de fazer, tendo a oportunidade de corrigir e, sobretudo, perceber onde o raciocínio inicial não estaria de acordo com os pressupostos do desejado, portanto, estamos a trabalhar uma das habilidades do pensamento computacional o teste e depuração.

Sessão 10 - Introdução ao Ozobot Blockly, exploração de conceitos básicos de programação. Nesta sessão, os alunos experienciam a programação do pequeno robô Ozobot através de blocos de programação visuais, recorrendo à plataforma OzoBlockly, descrita na secção 2.3.3. Todos os exercícios (**ANEXO E10**), são retirados dos guias (*Pacing Guide*) presentes na plataforma Ozobot Classroom. Na primeira parte da aula, os alunos aprendem o básico, de forma a navegar na plataforma OzoBlockly, a programar pequenos códigos simples de demonstração e a enviar o programa para o robô Ozobot via *Bluetooth*. Depois de consolidado o processo anterior, relembram os conceitos sobre ciências de computação, nomeadamente: sequências, ciclos e condições.

A plataforma de programação visual OzoBlockly, disponibiliza um conjunto limitado de idiomas de operação. Por esse facto foi usado o nível 2 de programação (*beginner*), em que os blocos de programação incluem imagens e texto na descrição, uma ajuda na barreira linguística. Usando a abordagem UMC é possível iniciar com tarefas simples e gradualmente ir introduzindo tarefas com maior complexidade de acordo com a compreensão dos alunos.

Estes exercícios pretendem desenvolver o raciocínio lógico e pensamento abstrato, na medida em que a programação é realizada com recurso a blocos de código (Programação Visual) o que implica a capacidade de prever o comportamento do robô a partir dos símbolos abstratos, bem como identificar se o robô está a executar o idealizado, de forma lógica e ordenada, ou seja, cada passo na programação

está associado a uma ação, onde cada aluno procura identificar a solução para um determinado problema/desafio.

3.5.2. Atividades complementares

O contexto do ambiente de aprendizagem no qual uma dada tecnologia é introduzida é tão relevante ou mais do que a tecnologia propriamente dita. Esta seção explora uma faceta importante: a participação dos diferentes intervenientes neste estudo num conjunto de atividades que estão interligadas com o objetivo desta investigação, com por exemplo o concurso *Bebras*.

Ao longo do ano letivo, fruto do plano anual de atividades ou de intervenções espontâneas, aconteceram momentos que importa referir e analisar, visto poderem contribuir com algum significado nesta investigação.

Em novembro de 2021, os alunos envolvidos nesta investigação participaram numa iniciativa internacional “Bebras – Castor Informático” (<http://bebras.pt>), que visa promover a informática e o pensamento computacional para alunos de todas as idades. É uma atividade desenhada para motivar os alunos, que não exige conhecimentos prévio, apenas algum pensamento lógico. A atividade consistiu na resolução de pequenos problemas que, de uma forma lúdica, estimulam o pensamento e o raciocínio, uma vez que os desafios apresentados neste concurso são divertidos, estimulantes e baseados em tarefas realizadas frequentemente por cientistas e engenheiros. Todos os desafios foram propostos por uma equipa da área das ciências da computação, composta por elementos de diferentes países.

Embora o objetivo principal do *Bebras* seja promover e encorajar o interesse dos alunos no pensamento computacional, várias iniciativas procuram utilizar o *Bebras* como instrumento de avaliação do pensamento computacional em diferentes contextos (Araujo et al., 2019; Chiazese et al., 2018; Lehtimäki et al., 2022).

Relativamente aos resultados obtidos, destacam-se quatro alunos no top 10% e quatro no top 25%, no que se refere ao 4.º ano, onze alunos ficaram situados no top 25%. Estes resultados são a nível nacional, sendo 2449 alunos do 3.º e 3070 do 4.º ano, num universo de 5519 alunos. Comparativamente, os alunos do 3.º ano obtiveram um resultado mais promissor. Contudo, convém referir que esta atividade foi realizada em novembro de 2021, portanto, fora do âmbito do programa de intervenção deste estudo.

Uma outra atividade, o teatro Robótica cidadania digital e Internet Segura, concretizada na XLI semana aberta do agrupamento e apresentada para toda a comunidade escolar. A peça de teatro focava as questões do *cyberbullying* e do facto de estarmos constantemente ligados, foram trabalhadas pelos alunos do 3.º e 4.º anos de escolaridade, sob a forma de um teatro colaborativo, no âmbito de OCPR. A peça foi integralmente escrita e encenada pelos cerca de 150 alunos, em articulação com os docentes titulares, os docentes de Programação e Robótica e o Clube de Programação e Robótica.

O entusiasmo dos alunos foi contagiante. As ideias foram deles e estavam bem cientes das questões relacionadas com esta temática. Funcionou muito bem porque não se limitaram a apresentar um filme, mas antes uma peça que participaram como principais protagonista e atores. Certamente que os conteúdos: sempre ligado; *cyberbullying*; partilha de dados ou a pegada digital estão agora mais presentes. Apresenta-se com uma forma diferente de aprender, que potencia benefícios pedagógicos e sociais em crianças destas idades.

No âmbito do projeto Erasmus+ ENNE (European National Networks for the Enhancement of Vocational Education and Training), o agrupamento iniciou uma atividade de partilha de práticas, experiências e sinergias, entre parceiros do projeto Erasmus+ ENNE. Entre elas encontra-se o Projeto de Programação e Robótica no 1.º ciclo, onde está inserido este estudo. As turmas envolvidas nesta investigação receberam os parceiros em sala de aula e partilharam o resultado de diferentes atividades relacionadas com a robótica educativa, programação e a literacia digital, de uma forma divertida e atrativa, com o lema “Programar a Brincar... Brincar a Programar”.

Durante o ano letivo, alguns alunos dos cursos profissionais de Técnico de Eletrónica, Automação e Computadores e Mecatrónica colaboraram na organização de atividades integradas na disciplina de OCPR, dirigida aos alunos dos 3.º e 4.º anos, da Escola EB2 e Escola EB1. A colaboração dos alunos do secundário envolveu a montagem de robôs de demonstração, experiências de eletricidade e eletrónica, criação de jogos em dispositivos programáveis, a monitorização das atividades realizadas e o apoio aos alunos do 1.º ciclo durante as atividades em sala de aula. Foram momentos de colaboração e articulação pedagógica, partilha de conhecimentos e aproveitamento de sinergias entre diferentes instituições do Agrupamento de Escolas. A interação entre pares é importante, na medida que confronta as crianças com outros pontos de vista e favorece o desenvolvimento socioafetivo e social.

No âmbito deste trabalho, o investigador fez parte do painel de uma sessão formativa/informativa (**ANEXO F**) através do centro de formação da ES, com o objetivo geral de promover Recursos Educativos Digitais (RED) de qualidade, de acesso livre, validados científica e pedagogicamente, e de apoiar o seu

desenvolvimento nas escolas da sua área geográfica. Através de uma sessão por videoconferência, o investigador partilhou, juntamente com as colegas responsáveis pela leção da disciplina de OCPR, o projeto Programação e Robótica, mais concretamente a intervenção ao nível do pensamento computacional no agrupamento de escolas. Iniciou a apresentação deixando claro que este projeto envolve uma articulação pedagógica entre diferentes docentes, acreditando que a força do digital não são as ferramentas, mas aquilo que fazemos com elas, nomeadamente as competências que são trabalhadas, as aprendizagens significativas, o desenvolvimento do raciocínio lógico e do pensamento abstrato, para além de outros, tais como a articulação curricular, o esbater/combater desigualdades no acesso à tecnologia, a mobilização de conhecimento em contextos reais, a resolução de problemas e o pensamento computacional.

Somos cada vez mais uma sociedade digital, e novas tecnologias estão ao dispor como a Robótica, Internet das Coisas, o *Big Data* e Inteligência Artificial/ *Machine Learning*. Mas todas elas só fazem sentido se forem usadas como um meio e não um fim.

Figura 19 - Apresentação sobre o PC



Fonte: Imagem criada pelo investigador, retirada da apresentação <https://tinyurl.com/2jbsy8er>

Ao nível do PC, foram abordados os seguintes temas: Uma moda, uma necessidade ou uma emergência; as diferentes formas de promover o PC, como é o caso da RE, atividades *offline/unplugged*, jogos de tabuleiro, programação; definição de PC; o porquê de introduzir o PC no ensino; que desafios para a escola; que competências é necessário promover; como articular e desenvolver múltiplas literacias; qual o *rationale* que está na base da integração do PC na educação; trabalho desenvolvido e teste de pensamento computacional; o Pensamento computacional, a programação e robótica educativa no currículo.

O objetivo desta sessão foi amplamente conseguido, foi possível proporcionar momentos de focos sobre o digital, problematização de escolhas, análise de estratégias, partilha de recursos e projetos e debate sobre “lugares-comuns” do discurso tecnológico.

3.6. Métodos e técnicas de recolha de dados

A recolha de dados é uma etapa muito importante em qualquer tipo de investigação e é muito comum encontrar na literatura sobre metodologia de investigação diferentes abordagens e classificações no que concerne aos elementos ligados à tarefa de recolha de dados.

Numa investigação qualitativa, o investigador tem acesso a diferentes métodos possíveis para recolher os dados, sendo a entrevista, a observação e o questionário as três formas privilegiadas deste tipo de investigação (Freixo, 2011, p. 192) admitindo, contudo, a existência de outros, mas derivados da observação.

Na presente investigação, o recurso a diferentes fontes de dados permitirá proceder à triangulação dos dados recolhidos, dando consistência aos dados, conferindo desta forma validade de construto, Yin (2015). Neste estudo, a aquisição dos dados pelo investigador foi concretizada através de observação direta, entrevista coletiva do tipo *focus group*, pré-teste e pós-teste. Os dados relevantes na implementação desta investigação foram recolhidos ao longo do 2.º e 3.º período do ano letivo 2021-2022.

3.6.1. Teste de Pensamento Computacional

Com a crescente importância do PC em todos os níveis de ensino, é essencial ter avaliações válidas e confiáveis. De acordo com o estudo realizado por Ramos e Espadeiro (2014), já nessa altura, os autores evidenciavam a importância de instrumentos de avaliação adequados à avaliação do pensamento computacional, por forma a criar as condições necessárias de integração curricular. Mais recentemente, Relkin e Bers (2021) consideram que a avaliação do PC é importante para documentar o progresso da aprendizagem, aferir a eficácia das aulas, contribuir para o desenvolvimento do currículo e para a identificação de alunos com dificuldades de aprendizagem.

Uma das preocupações iniciais do investigador centrou-se em conseguir encontrar um instrumento que possibilitasse medir o nível de desenvolvimentos do PC dos alunos. Por conseguinte, foi feito um levantamento dos instrumentos já existentes que fossem ao encontro de um dos objetivos desta

investigação “Identificar os níveis de desenvolvimento do Pensamento Computacional”. Da literatura encontrada, foram selecionados dois testes de pensamento computacional, *TechCheck-2*, desenvolvido ao longo dos anos pelo *DevTch Research Group*, liderado pela investigadora Marina Bers. Este teste é composto por 15 questões/desafios com atividades *unplugged*, com conteúdos básicos das ciências da computação, destinados a crianças entre os 7 a 9 anos. Não exige experiência em codificação/programação (Relkin & Bers, 2021).

O segundo teste foi desenvolvido por Marcos González, em 2016, no âmbito da sua investigação de Doutorado, com a finalidade de colmatar a ausência de instrumentos de avaliação e medir as habilidades do PC (González, 2016). Diferentemente de outras propostas, o seu teste de PC passou por um rigoroso processo de validação, sendo referenciado em vários trabalhos e aplicado em diferentes estudos (Brackmann, 2017; Román-González et al., 2017; Zhao & Shute, 2019), e em Portugal na investigação de Marques (2021).

A escolha acabou por recair no teste de Marcos González, por se tratar de um instrumento que já tinha respondido aos mesmos objetivos, mas também por possibilitar comparações dos resultados em diferentes situações. Além disso é de fácil aplicação e com um processo de validação formal que permite atingir um resultado mais preciso e confiável (Marques, 2021).

O referido teste, é composto por 28 questões com 4 alternativas de resposta (A, B, C e D), das quais apenas uma é a correta. Baseado nas normas da CSTA, o teste tenta identificar habilidades na resolução de problemas, baseando-se nos conceitos básicos das ciências da computação, em sintaxes lógicas usadas na programação (variáveis, sequências simples através das direções, condições através de operações condicionais simples e compostas, ciclos e funções) e inclui dimensões do Pensamento Computacional como a decomposição, a abstração, o reconhecimento de padrões, algoritmos e depuração. Uma relação das dimensões do PC com as questões e os conceitos básicos das ciências das Ciências da Computação é apresentada na tabela 9, com base nos estudos de Brackmann (2017) e González (2016).

Tabela 9 - Conceitos de Ciências da Computação e de Pensamento Computacional

Questão	Conceitos de Ciências de Programação							Dimensões do Pensamento Computacional					
	Direções		Loop		Condições		Repetição	Funções	Abstração	Decomposição	Reconhecimento Padrões	Algoritmos	Depuração
	1.	2.	3.	4.	5.	6.	7.	A.	B.	C.	D.	E.	
1	X							X			X	X	
2	X							X			X	X	
3	X							X			X	X	
4	X								X	X	X	X	

5		X							X	X	X	X
6		X							X	X	X	X
7		X						X	X	X	X	X
8		X								X	X	X
9			X							X	X	X
10			X						X	X	X	X
11			X					X	X	X	X	X
12			X					X	X	X	X	X
13				X				X	X		X	X
14				X				X		X	X	X
15				X				X	X	X	X	X
16				X							X	X
17					X					X	X	X
18					X					X	X	X
19					X						X	X
20					X					X	X	X
21						X		X	X		X	X
22						X		X	X	X	X	X
23						X		X	X	X	X	X
24						X					X	X
25							X	X	X	X	X	X
26							X	X	X	X	X	X
27							X	X	X	X	X	X
28							X	X	X	X	X	X
1 - Sequências simples 2 - Ciclos de repetição por número de vezes (<i>repeat x times</i>) 3 - Ciclos de repetição até uma determinada condição (<i>repeat until</i>) 4 - Condição simples (<i>if</i>) 5 - Condição composta (<i>if/else</i>) 6 - Ciclo de repetição enquanto uma condição se verificar (<i>while</i>) 7 - Funções simples, de modo a repartir o problema e agrupar tarefas	A - Abstração B - Decomposição C - Reconhecimento de padrões D - Algoritmos E - Depuração											

Fonte: Tabela elaborada pelo investigador, adaptado de (Brackmann, 2017; González, 2016)

Cada pergunta deste teste, aborda um ou mais conceitos básicos de programação, aumentando a sua complexidade de forma gradual. A partir do momento em que o aluno inicia o teste, tem 45 minutos para o concluir. No entanto, não é obrigatório responder a todas as questões. Em termos gerais, a estrutura do teste é composta por 7 conceitos básicos sobre programação, de acordo com a tabela anterior, nomeadamente:

- 4 questões de sequências simples (1 a 4);
- 4 questões sobre ciclos de repetição por número de vezes (5 a 8);
- 4 questões sobre ciclos de repetição até uma determinada condição (9 a 12);
- 4 questões por condição simples: *if* (13 a 16);
- 4 questões por condição composta: *if/Else* (17 a 20);
- 4 questões por ciclo de repetição: *while* (21 a 24);
- 4 questões com funções simples (25 a 28).

Antes de iniciar o teste, são explorados três tipos de questões, para que os alunos se familiarizem com o tipo de questão e com as personagens envolventes. No primeiro tipo de questão exemplificativa, é apresentada uma sequência de setas, onde os alunos são questionados sobre os passos necessários para levar o “Pac-Man” até ao fantasma pelo caminho assinalado. Este exemplo é apresentado na figura seguinte.

Figura 20 - Exemplo I da questão do Teste de PC

Exemplo I

Neste primeiro exemplo perguntam-te quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado.

A opção correta neste exemplo é a B. Assinala a resposta correta, na tabela que está por baixo da pergunta.

<p>Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?</p>	<p>Opção A</p> <p>→ → ↓</p>
	<p>Opção B</p> <p>→ → ↑ ✓</p>
	<p>Opção C</p> <p>→ ↑ ↑</p>
	<p>Opção D</p> <p>→ ↓ ↓</p>
<p>Exemplo I <input type="checkbox"/> A <input checked="" type="checkbox"/> B <input type="checkbox"/> C <input type="checkbox"/> D</p>	

Fonte: Imagem retirada do Teste de PC, (Marques, 2021)

No segundo exemplo, os alunos são novamente questionados sobre os passos necessários para levar o “Pac-Man” até ao fantasma pelo caminho assinalado, mas agora através uma sequência de blocos de código em vez das setas de direção. Este exemplo é apresentado na figura seguinte.

Figura 21 - Exemplo II da questão do Teste de PC

Exemplo II

Neste segundo exemplo perguntam-te, novamente, quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado. Mas neste caso as opções de resposta, em vez de serem setas, são blocos/instruções que encaixam uns nos outros.

A opção correta neste exemplo é a C. Assinala a resposta correta, na tabela que está por baixo da pergunta.

<p>Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?</p>	<p>Opção A</p> <p>mover para a frente virar à esquerda mover para a frente mover para a frente</p>	<p>Opção B</p> <p>mover para a frente virar à direita mover para a frente mover para a frente</p>	
	<p>Opção C</p> <p>mover para a frente ✓ mover para a frente virar à esquerda mover para a frente</p>	<p>Opção D</p> <p>mover para a frente mover para a frente virar à direita mover para a frente</p>	
	<p>Exemplo I <input type="checkbox"/> A <input type="checkbox"/> B <input checked="" type="checkbox"/> C <input type="checkbox"/> D</p>		

Fonte: Imagem retirada do Teste de PC, (Marques, 2021)

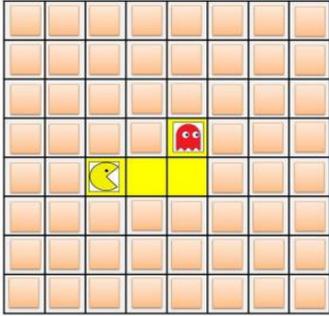
O terceiro tipo de questão, está relacionada com o personagem Artista que desenha diferentes figuras. Os alunos têm de identificar o bloco de código/instruções que permite esboçar essas figuras. Nestes exemplos é sempre indicado onde o Artista inicia os seus desenhos, por uma pequena seta de direção a cinza. O exemplo deste tipo de questão é apresentado na figura seguinte.

Figura 22 - Exemplo III da questão do Teste de PC

Exemplo II

Neste segundo exemplo perguntam-te, novamente, quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado. Mas neste caso as opções de resposta, em vez de serem setas, são blocos/instruções que encaixam uns nos outros.

A opção correta neste exemplo é a C. Assinala a resposta correta, na tabela que está por baixo da pergunta.

<p>Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?</p> 	<p>Opção A</p> <ul style="list-style-type: none"> mover para a frente virar à esquerda ▾ mover para a frente mover para a frente 	<p>Opção B</p> <ul style="list-style-type: none"> mover para a frente virar à direita ▾ mover para a frente mover para a frente
	<p>Opção C</p> <ul style="list-style-type: none"> mover para a frente ✓ mover para a frente virar à esquerda ▾ mover para a frente 	<p>Opção D</p> <ul style="list-style-type: none"> mover para a frente mover para a frente virar à direita ▾ mover para a frente

Exemplo I A B C D

Fonte: Imagem retirada do Teste de PC, (Marques, 2021)

A versão original do teste foi desenvolvida em Espanhol (idioma europeu), por Marcos Román González e traduzido por Christian Puhmann Brackmann para Português (idioma Brasileiro) com autorização do seu autor (Prof. Dr. Marcos Román González) que, também autorizou que o seu teste fosse utilizado nesta investigação (**ANEXO A**).

A tradução e adaptação do teste para a língua portuguesa, está patente no trabalho de António Soares Marques, na sua tese de doutoramento. No estudo de António Marques, é feita referência a todo o processo meticoloso de adaptação, elaboração e validação do instrumento para a língua portuguesa. O produto final apresentou-se pela sua simplicidade e acessibilidade e com uma linguagem adaptada, sendo passível de ser aplicado a crianças que frequentem o 3.º e 4.º anos de escolaridade (Marques, 2021).

De forma, a agilizar o processo de aplicação do teste, o investigador adaptou, cuidadosamente, o teste para uma plataforma on-line Quizizz (<https://quizizz.com/>), que fosse ao encontro da faixa etária dos alunos, para concretizarem o teste através de um *tablet* com ligação à internet.

A pretensão, neste passo, foi manter ao máximo a estrutura do teste, bem como as suas personagens, uma vez que a sua transformação arriscaria alterar o interesse/motivação demonstrado pelos alunos, o que poderia modificar os resultados, principalmente no que concerne aos níveis de concretização. De salientar que a motivação dos alunos para este tipo de perguntas/exercícios é, por norma, um fator

decisivo, o que exige que os programas sejam próximos da realidade vivenciada pelos alunos e que tenham temas agradáveis.

De acordo com alguns estudos (Safarati & Rahma, 2020; Sousa & Araújo, 2021), a plataforma adotada, evidência potencialidades diagnosticadas relativamente ao uso: compromisso dos alunos por meio de questionários interativos e grafismos apelativos; promoção da motivação devido aos elementos de *gaming* como competição, pontuação, desafios, *feedbacks* e *ranking*; favorecimento de discussões sobre o conteúdo exposto; possibilidade de tratamentos de estatística simples.

3.6.2. Observação por observação direta

Sendo a intervenção deste estudo uma investigação qualitativa, uma das técnicas de recolha de dados usada foi a da observação. Sendo uma das técnicas mais antigas, esta faz-nos perceber os comportamentos no seu contexto real.

Segundo Gil (1999), as observações permitem obter um registo dos acontecimentos à medida que estes vão tendo lugar no tempo, constitui a forma mais apropriada de conhecer a realidade, uma vez que permite ao investigador uma melhor compreensão do caso visto que não tem qualquer controlo na situação com que se depara.

Neste trabalho, o investigador assumirá o papel de participante observador e a observação será desenvolvida no contexto da sala de aula, de acordo com as estratégias pré-definidas no grupo de trabalho. Caracterizar-se por uma “observação participante”, que segundo Yin (2015), citado em Meirinhos e Osório (2010), “é um modo especial de observação, em que o investigador não é meramente um observador passivo, mas pode assumir uma variedade de papéis no estudo de caso, podendo mesmo participar em acontecimentos a serem estudados”.

A decisão sobre o tipo de observação foi uma das primeiras necessidades. Dependeu dos objetivos a atingir, do grupo a observar, do tempo disponível para a recolha de dados e do contexto a observar. Assim, nesta investigação, optou-se por praticar uma observação direta. Mas a incompatibilidade horária do investigador e o facto de não ser um professor da disciplina de OCPR nem sempre permitiu esse modo de observação, tendo-se apenas verificado apenas o contacto direto com uma das turmas intervenientes na investigação.

Sempre que houve a possibilidade de o investigador realizar uma observação direta em sala de aula, esta foi concretizada com as professoras cooperantes responsáveis pela disciplina de OCPR, no desenvolvimento dos projetos, de forma a registar os procedimentos dos alunos e o apoio das professoras

na implementação das atividades. Durante esse tempo, foram respeitados os princípios éticos do consentimento informado e da confiabilidade, sendo garantida aos participantes a confidencialidade e anonimato.

3.6.3. Inquéritos por entrevista coletiva do tipo *focus group*

De acordo com Morgan e Kruger (1998), a entrevista coletiva do tipo *focus group*, é uma técnica de recolha de dados, de natureza qualitativa e interpretativa do discurso. Pode ser aplicada em qualquer fase da investigação, podendo ser combinada com outras técnicas. Trata-se de uma estratégia de recolha de dados que tem objetivos muito específicos e, por isso, deve ser considerada como uma técnica autónoma para a recolha de dados nas Ciências Sociais e Humanas.

Na perspetiva dos autores, a entrevista coletiva do tipo *focus group*, é uma técnica de investigação de recolha de dados sobre um tópico apresentado pelo investigador, que através da interação de um grupo de pessoas, permite o registo de experiências e das reações dos indivíduos participantes, o que não seria possível de captar por outras técnicas de recolha de dados, como por exemplo, a observação participante, as entrevistas ou os questionários. Esta definição, segundo os autores, assenta em três pilares fundamentais: os *focus group* são um método de investigação direcionado à recolha de dados; foca-se na interação entre os diferentes elementos do grupo como fonte de dados e reconhece a importância do papel ativo do investigador na dinamização/condução da discussão.

Por sua vez Wilkinson (2004), considera o *focus group* como uma discussão informal que ocorre entre um conjunto de indivíduos previamente selecionados, discussão esta centralizada num determinado tópico, tendo como principal objetivo descrever e compreender os significados e interpretações que o grupo selecionado tem sobre um determinado assunto em particular.

A contextualização teórica acabada de realizar torna clara a escolha do *focus group* enquanto recurso para a investigação. De facto, assumindo uma parte da investigação um carácter exploratório e tendo como objetivo proceder a uma recolha de dados sobre a opinião de um conjunto de pessoas (professoras titulares e professoras cooperantes de OCPR do 3.º e 4.º anos) em relação a um tema específico (Robótica Educativa e Pensamento Computacional), o recurso ao *focus group* surgiu como uma das opções a seguir.

Por outro lado, a realização de um *focus group*, contempla um conjunto de princípios orientadores que é importante ter em conta e que diz respeito à constituição do grupo de participantes, à duração das

sessões e aos cuidados a ter na transcrição das discussões. Krueger e Casey (2015), enumeram uma lista de requisitos a ter em conta na utilização da entrevista coletiva do tipo *focus group*:

- o número de participantes deverá estar compreendido entre os 5 a 9 elementos. Neste estudo estiveram presentes 8 docentes;
- a constituição dos grupos deve ser uniforme. As professoras entrevistadas foram selecionadas pelo critério de lecionação (3.º e 4.º ano) das turmas intervencionadas;
- a duração das sessões não deve ultrapassar as duas horas. Neste estudo, a duração foi aproximadamente de 1h15, tempo suficiente para que todos os entrevistados se exprimissem e sem ser cansativo;
- a tema da sessão deve ser um tópico de interesse comum para todos os elementos constituintes do grupo. As questões orbitaram em torno dos temas da aprendizagem da Robótica Educativa e do PC em crianças do 1.º ciclo, assuntos atuais e pertinentes, capazes de suscitar debates e reflexões sobre a importância da RE no desenvolvimento de um conjunto de atitudes e habilidades relacionadas com Pensamento Computacional;
- a concretização de um *focus group*, deve ser assegurado por um moderador. Neste estudo, o investigador orientou/moderou a sessão, promoveu a participação e a interação de todos as professoras, e assegurou que não existiu dispersão.

Relativamente ao guião do *focus group*, as questões elaboradas foram revistas e analisadas pelo orientador desta tese de mestrado e pelas colegas cooperantes responsáveis pela disciplina de OCPR **(ANEXO G)**.

Neste estudo, a técnica da entrevista coletiva do tipo *focus group* vem complementar outras técnicas (inquérito pré-teste e pós-teste), através da qual se procura atingir objetivos previamente estabelecidos, integrando em si técnicas como a entrevista não direcionada, a observação e a análise do discurso. De facto, Creswell (2014) considera que as entrevistas que envolvem questões não estruturadas permitem uma maior liberdade de ação. No entanto, são centradas num assunto preciso e destinadas a suscitar conceções e opiniões dos participantes.

A entrevista aos professores teve como propósito conhecer a opinião destes sobre as aulas de Programação e Robótica no 1.º Ciclo (3.º e 4.º anos), nomeadamente: qual a importância da RE e da programação na aprendizagem dos alunos e no nível de consolidação de conhecimentos; de que forma a RE promoveu as dimensões do PC; qual é a avaliação da metodologia aplicada nas aulas de OCPR; como caracterizam os materiais de RE adotados durante as aulas de OCPR.

As professoras foram convidadas inicialmente de forma presencial, sendo-lhes posteriormente endereçado via email um convite formal. Para facilitar a leitura e a compressão, foi anexado o guião das questões do inquérito por questionário do tipo *focus group*.

Durante a fase de entrevista, o investigador atuou como moderador e orientou a sessão abordando todas as questões previstas no guião. A sessão decorreu em aproximadamente de 1h15, tempo sido considerado pelos participantes um período de tempo razoável. Todas as questões, foram avaliadas relativamente à clareza, dificuldade, incerteza e omissão de algum tópico considerado importante.

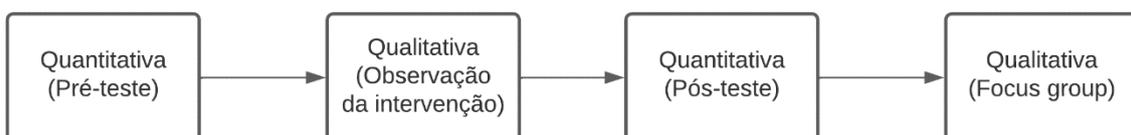
Importa lembrar que esta entrevista por *focus group* teve um carácter exploratório e pretendia auscultar a opinião de um grupo de professoras sobre a utilização da RE na disseminação do PC em contexto educativo.

3.7. Métodos e técnicas de análise de dados

Para análise de dados recolhidos neste estudo, o investigador privilegiou a análise mista de dados qualitativos e quantitativos, baseando-se no princípio central da complementaridade, onde será possível recolher, analisar e integrar ou relacionar dados qualitativos num único estudo, ou em diversas fases da mesma investigação (Creswell, 2014).

Ainda neste contexto de análise mista, uma forma de articular uma estratégia quantitativa com uma qualitativa é o caso apresentado por Miles e Huberman (1994) que sugere quatro tipos de planos de pesquisa que permitem integrar as duas abordagens no mesmo plano. Com base nesses planos de pesquisa e adapta-se, para o contexto deste estudo, uma proposta para integração da investigação qualitativa e quantitativa que poderá ser esquematizada na figura seguinte.

Figura 23 - Plano de pesquisa para integração da investigação mista



Fonte: Elaborado pelo investigador com adaptação de Miles e Huberman

Conforme ilustrado, o plano inicia com um método quantitativo por questionário (Teste de Pensamento Computacional), seguido de uma observação do estudo experimental no terreno (intervenção com sessões de trabalho junto dos alunos), como fase intermédia, seguindo-se um estudo por questionário

(pós-teste), para testar os resultados da intervenção feita no passo anterior e, finalmente uma análise de conteúdo complementar que acrescenta profundidade aos resultados obtidos nos passos anteriores.

Pesquisas desta natureza têm como objetivo analisar os dados recolhidos e sistematizar as informações obtidas na pesquisa por meio de questionários (teste por questionário) e entrevistas (*focus group*).

O estudo de conteúdo foi um método utilizado para análise do texto proveniente da gravação da entrevista coletiva por *focus group*, como refere Bardin (2016). Trata-se de uma técnica de análise das comunicações, que originam dados textuais dos quais é preciso extrair sentido, que pode ser adaptado aos mais diversos campos da comunicação (p.33).

Segundo o mesmo autor, a análise de conteúdo de tipo exploratório realiza-se em três momentos sucessivos: a pré-análise, a exploração do material; e o tratamento dos resultados (a inferência e a interpretação). A pré-análise é o momento de organização do material, de escolha dos documentos a serem analisados, de formulação de hipóteses ou questões de investigação, de elaboração de indicadores que alicerces a interpretação final. Na exploração do material, são realizadas as decisões tomadas na pré-análise em que o investigador organiza os dados e os transforma através das operações de codificação, enumeração e categorização de acordo com o quadro teórico de referência. Ao processar os resultados obtidos e as interpretações, é preciso comparar enunciados e ações entre si, para ver se existe um conceito unificador, de forma a serem apreciáveis e eficazes, em tabelas de resultados, esquemas e gráficos que revelem informações importantes (p.123).

Na análise de dados provenientes dos testes de Pensamento Computacional (pré-teste e pós-teste), recorreu-se a uma análise quantitativa baseada na estatística básica descritiva, em medidas numéricas e gráficos básicos que irão resumir o comportamento das variáveis estudadas, com objetivo de descrever e resumir os dados obtidos.

Segundo Reis (1996, p. 15), a estatística descritiva consiste na recolha, análise e interpretação de dados numéricos através da criação de instrumentos adequados: quadros, gráficos e indicadores numéricos. Pretende proporcionar relatórios que apresentem informações sobre a tendência central e a dispersão dos dados. Para tal, deve-se evidenciar: valor mínimo, valor máximo, soma dos valores, contagens, média, moda, mediana, variância e desvio padrão.

Os dados quantitativos, provenientes do teste de Pensamento Computacional aqui expostos foram tratados através do *software* IBM SPSS – versão 28.0, para ambiente Windows, com licença disponibilizada pela Escola Superior de Tecnologia e Gestão do Politécnico do Porto.

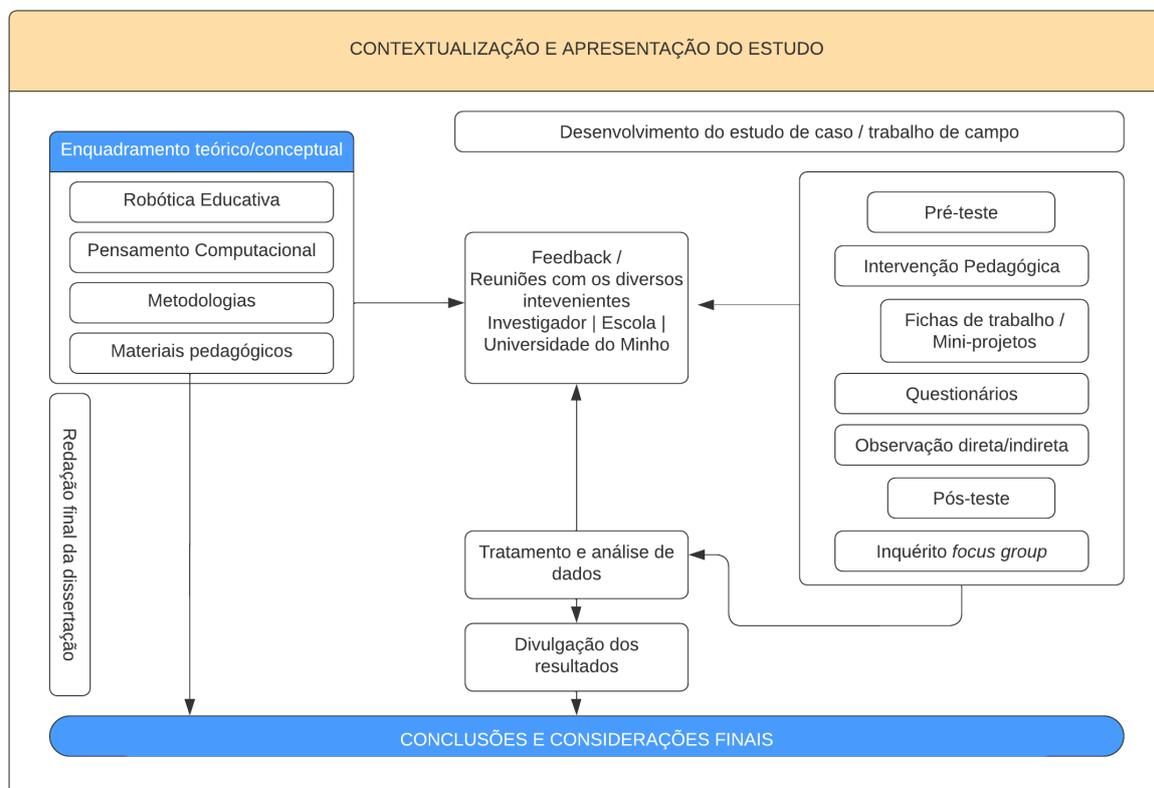
Facilmente compreende-se que o tratamento e a análise de dados são de extrema importância para a investigação, quer numa perspetiva dedutiva (veracidade das hipóteses) quer na perspetiva indutiva (ferramenta utilizadas).

Nesta investigação, ao estudar a utilização de múltiplas fontes de evidência (observação, teste por questionário e *focus group*), procurou-se analisar os dados recolhidos e sistematizar as informações obtidas na pesquisa.

3.8. Calendário de atividades

Nesta secção, apresenta-se um mapa sintético das atividades planeadas para os alunos, no âmbito deste trabalho, bem como a sua calendarização. Tenta-se assim, representar de forma esquemática o processo de desenvolvimento da investigação. Este “mapa mental” permite ter uma visão helicóptero sobre o trabalho realizado.

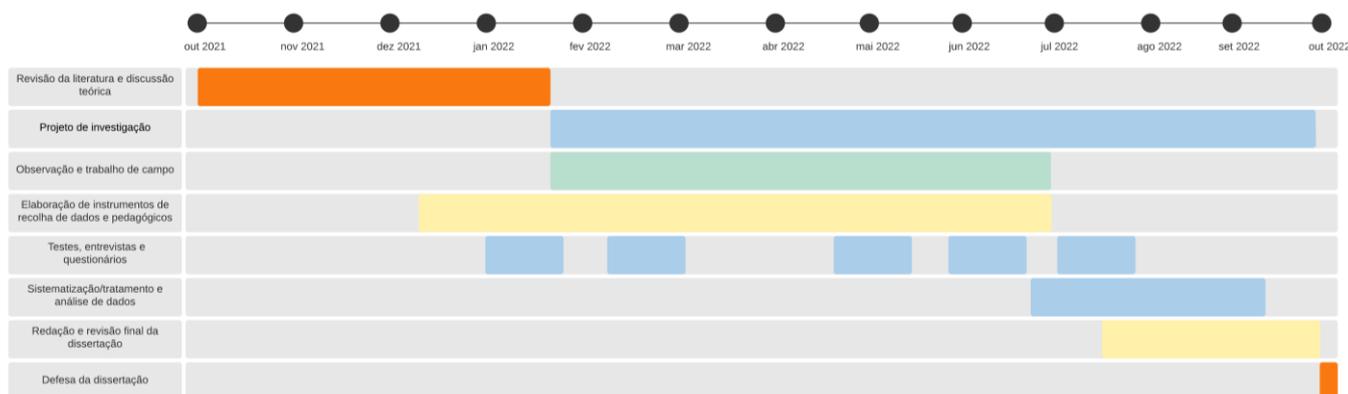
Figura 24 - Estrutura de desenvolvimento do estudo



Fonte: Elaborado pelo investigador

De igual modo, na figura seguinte é apresentado um cronograma que indica a sequência de atividades realizadas e os momentos em que cada uma delas foi executada.

Figura 25 - Cronograma



Fonte: Elaborado pelo investigador em novembro de 2021

3.9. Validade, fiabilidade e confiabilidade

Esta investigação tem como finalidade obter respostas válidas e fiáveis para os objetivos definidos na secção 1.3. Foi desenvolvida num contexto do qual o investigador fez parte, no entanto, não diretamente como professor dos alunos das turmas intervencionadas, mas como professor em modo artificial, conhecedor da situação em estudo por dentro (em anos anteriores já lecionou a disciplina de OCPR), como professor do agrupamento, sendo-lhe possível interferir, ainda que de modo indireto, na intervenção pedagógica, na situação em análise. Por outras palavras, não se limitou ao papel de mero espetador. Tendo em conta esta situação, a imparcialidade e a independência foram princípios que sempre nortearam o investigador.

A qualidade de investigação científica foi uma preocupação do investigador durante todo o processo de investigação, normalmente associada à validade, fiabilidade e confiabilidade dos instrumentos de recolha de dados.

Para Yin (2015), a qualidade de um estudo de caso está relacionada com critérios de validade e fiabilidade. A “Validade de Construto” verifica até que ponto uma medida utilizada num estudo de caso é adequada aos conceitos a serem estudados. A “Validade Externa” mostra até que ponto as conclusões de um estudo de caso podem ser generalizáveis a outras investigações de casos semelhantes (i.e. até que ponto os resultados podem ser transferidos para outros contextos). A “Validade Interna” avalia em que medida o investigador demonstrou a relação causal entre dois fenómenos observados (i.e. até que ponto os resultados recolhidos representam os dados obtidos). A fiabilidade de um estudo de caso mostra em que medida outros investigadores chegariam a resultados idênticos, utilizando as mesmas metodologias na mesma investigação.

Segundo Coutinho (2021), a validade consiste na certificação de que os instrumentos utilizados garantem que se atinjam resultados coerentes e passíveis de serem aceites numa determinada investigação, enquanto a fiabilidade é a certificação de que os dados recolhidos correspondem à realidade (i.e. garantem que outro investigador poderá realizar uma investigação semelhante e obterá resultados idênticos). Apesar de diferentes, estes dois conceitos estão intimamente ligados e só a conjugação destas duas certificações permitem que todo o trabalho de investigação desenvolvido apresente utilidade para a comunidade e espelhe a competência do investigador no seu papel.

A autora sublinha que devemos ser capazes de verificar se os instrumentos de recolha de dados, utilizados na investigação, nos dão garantias da qualidade informativa desses. Afirma também que, é “fundamental que todo o investigador em Ciências Sociais e Humanas se preocupe com a questão da fiabilidade e validade dos métodos a que recorre, sejam eles de cariz quantitativo ou qualitativo” (Coutinho, 2021, p.231).

Quanto à validade dos instrumentos de recolha de dados usados nesta investigação, foram realizados inicialmente dois: o teste de pensamento computacional e o inquérito por entrevista do tipo *focus group*. Como já referido na secção 3.6.1, o teste de PC foi amplamente testado e validado, possibilitando comparações dos resultados para que se possa atingir um resultado mais preciso e confiável. No entanto, antes de aplicar o teste, foi realizado um trabalho prévio no sentido de aferir se a funcionalidade da plataforma quizizz permitiria a implementação do teste.

A aplicação do teste pode ser realizada através de um documento impresso ou via digital, através da internet utilizando um *web browser* (Chrome, Firefox, Edge) e acedido de forma virtual através de qualquer dispositivo digital (p. exe. computador, *tablet*, telemóvel). Optou-se pela versão digital por questões de logística e aquisição de dados. Apesar de o teste estar disponível através de um formulário (*google forms*) em versão brasileira, todas as questões foram convertidas/adaptadas para a plataforma quizizz, numa versão portuguesa, com base no trabalho de António Marques, já referido na secção 3.6.1.

Este teste piloto com a plataforma quizizz, serviu para aferir as condições e procedimentos no dia da aplicação do pré-teste e garantir a validade e fiabilidade do instrumento de recolha de dados. A professora titular e os alunos do 3.º ano de escolaridade que participaram na sessão piloto, fazem parte de uma escola pertencente ao Centro de Formação de Associação de Escolas (CFAE) onde está inserido o agrupamento deste estudo. Tal afigura-se como condição essencial para a criação de um meio semelhante ao do estudo. Esta turma do 3.º ano de escolaridade, com 21 alunos, não tem qualquer

oferta educativa na matriz curricular, nem desenvolveu atividades pedagógicas ligadas à robótica ou programação.

Nesta intervenção piloto, foram testadas diversas condições, relativamente ao uso da plataforma quizizz, nomeadamente:

- condições relativas aos recursos técnicos/físicos (um computador/*tablet* por aluno, acesso à internet, disposição em sala de aula, resposta da plataforma em tempo real);
- Tempo de resolução do teste;
- Perceção da usabilidade da plataforma;
- Motivação e autonomia dos alunos;
- Dificuldades evidenciadas pelos alunos.

Durante a intervenção piloto, o investigador anotou os comportamentos e comentários dos alunos e da professora titular de turma, existindo um cuidado em transmitir permanentemente um *feedback* construtivo, com o objetivo de auxiliar na execução das respostas, motivando o aluno para responder de acordo com a sua perceção do tema e para colocar em prática as suas competências. Foi-lhes explicado que, no âmbito deste estudo, o importante é participar, destacando o facto de esta ser uma atividade extra que não iria interferir nos conteúdos abordados, nem considerada para avaliação.

Figura 26 - Aplicação do Teste piloto



Fonte: Elaborado pelo investigador

As principais anotações retiradas desta pequena intervenção, foram:

- Necessidade de criar um *link* no ambiente de trabalho de computador/tablet, que permitisse abrir o teste de uma forma célere;
- Verificou-se que os alunos não apresentaram dificuldades na usabilidade da plataforma;
- A plataforma permite o uso de *zoom* nas imagens, o que auxilia na compreensão e diferenciação das personagens e dos exercícios;
- A explicitação das instruções iniciais deve ser feita em conjunto com os alunos pelo aplicador do teste (professora de OCPR ou investigador), exemplificando com os três tipos de questões, de forma que os alunos percebam o objetivo e dinâmica do teste;
- Nas primeiras questões (1 a 10), os alunos compreenderam o objetivo e no geral conseguiram responder;
- Nas questões que responderam, o tempo médio de resposta está dentro do espectável (1,5 minutos por questão / em média).

Todas as dificuldades e comentários dos alunos, bem como as reações, foram registadas através de notas de campo (descritivas e reflexivas), tendo especial atenção aos diálogos, atitudes e ações. A plataforma quizizz demonstrou a confiança necessária para ser adotada como suporte à validação empírica.

Safarati & Rahma (2020) e Sousa & Araújo (2021), sugerem que a adoção do quizizz promove um maior envolvimento dos estudantes na aprendizagem, favorece o feedback aos alunos e incentiva a um ensino mais reflexivo de perguntas e respostas baseado no conceito de gamificação.

Relativamente ao instrumento de recolha de dados, inquérito por entrevista do tipo *focus group*, procurou-se obter dados da perceção dos diferentes participantes, sobre a temática em estudo, descrita na secção 3.6.3. Este inquérito serviu ainda para adquirir informação que não foi possível obter através da observação, triangulando as diferentes opiniões e interpretações dos professores titulares e de OCPR.

O guião de *focus group* foi validado através da interação do investigador com o seu orientador, considerando os objetivos direcionados para a realização do projeto de investigação em geral e do *focus group* em particular, bem como a estrutura do guião da entrevista, participantes, as condições logísticas de realização e o local.

Segundo Coutinho (2021), a investigação quantitativa é conseguida com recurso a critérios de validade e fiabilidade. No paradigma qualitativo, faz-se procurando a confiabilidade (p.237).

Neste estudo, conforme descrito, foi preocupação do investigador adotar procedimentos de forma que os resultados obtidos fossem confiáveis e refletissem a realidade apresentada por todos os participantes num determinado contexto.

3.10. Questões éticas

Durante a realização deste estudo, estiveram presentes os princípios éticos quanto à proteção de dados dos participantes na investigação e das instituições envolvidas. Manteve-se uma preocupação constante com o cumprimento do Regulamento Geral de Proteção de Dados (RGPD) (UE) 2016/679, de 27 de abril de 2016, destinado a regulamentar o direito à privacidade e proteção de dados pessoais a todos os indivíduos da União Europeia e do Espaço Económico Europeu. No que concerne aos participantes, prevaleceu um cuidado sobre a informação previa quanto aos termos da intervenção dos inquiridos neste estudo. Foram informados, via email pela professora titular, acerca da natureza desta investigação, dos objetivos, dos instrumentos de inquirição, da durabilidade do estudo e foram prestados esclarecimentos sobre os dados recolhidos.

Para informar e requerer o pedido de autorização aos Encarregados de Educação (**ANEXO C**), recorreu-se ao professor titular de turma que estabeleceu o contacto e os informou/esclareceu sobre a natureza e objetivos da investigação. Houve uma constante preocupação em respeitar o direito de privacidade, descrição e anonimato de todos os participantes, certificando-os que os dados fornecidos são totalmente confidenciais.

Os Encarregados de Educação foram informados e deram autorização para que os seus educandos participassem no respetivo estudo.

Relativamente aos professores participantes do *focus group*, subsistiu a preocupação em solicitar autorização para a gravação áudio das entrevistas, de forma a proteger a imagem e recolha de dados, prevalecendo o anonimato.

Após a construção dos instrumentos, foi necessário solicitar o pedido de autorização para aplicação de inquéritos/realização de estudos de investigação em meio escolar à Direção-Geral de Estatística da Educação e Ciência, de acordo com o previsto no despacho n.º 15847/2007, publicado no DR 2.ª série n.º 140, de 23 de julho, não sendo exequível a realização de qualquer estudo de investigação em meio escolar, sem a solicitação através de um pedido formal e, posterior feedback da tutela com uma apreciação positiva.

Este pedido foi concretizado através da plataforma (<http://mime.dgeec.mec.pt>) disponível na página da Direção-Geral de Estatística da Educação e Ciência, criada para agilizar este processo, designada por Monitorização de Inquéritos em Meio Escolar (MIME).

Para concretizar este pedido foi necessário fazer o registo/inscrição da entidade, especificando o âmbito desta investigação e anexar, para aprovação, um conjunto de requisitos, nomeadamente: proposta de autorização para os encarregados de educação, a descrição do estudo, o seu enquadramento, notas metodológicas, instrumentos de inquirição, declaração do orientador da investigação, entre outras.

A autorização (**ANEXO B**) foi concebida para que a recolha fosse feita até 29 de julho de 2022, a 139 alunos do 3.º e 4.º anos de escolaridade, circunstâncias que foram escrupulosamente cumpridas. Com a aprovação, foi comunicado por email algumas informações/obrigações, das quais:

- A realização dos Inquéritos fica sujeita a autorização das Direções dos Agrupamentos de Escolas do ensino público a contactar para a realização do estudo.
- Merece especial atenção o modo, o momento e condições de aplicação dos instrumentos de recolha de dados em meio escolar, devendo fazer-se em estreita articulação com as Direções dos Agrupamentos e com os encarregados de educação ou quem tutele os menores inquiridos.
- O cumprimento do Regulamento Geral de Proteção de Dados, garantindo o anonimato dos participantes, não identificabilidade, confidencialidade, proteção e segurança dos dados pessoais a recolher e tratar no presente estudo. Assim, dada a natureza dos dados pessoais deve ser dado conhecimento a todos os inquiridos e a quem intervenha na recolha e tratamento de dados.
- As declarações de consentimento a utilizar, bem como os instrumentos de inquirição, devem estar disponíveis na recolha do prévio consentimento dos representantes legais dos alunos, de modo inequívoco, informado e esclarecido, junto dos inquiridos, titulares dos dados, no caso de menores, junto dos encarregados de educação ou quem tutele os menores inquiridos.

A obtenção desta autorização é de extrema importância, pois legalmente, sem a mesma, este estudo não poderia ser concretizado nem publicitado.

Para além das principais preocupações e procedimentos descritos anteriormente de forma a assegurar que a investigação se desenvolvesse com respeito pelos participantes, foi considerada informação adicional disponibilizada no site (<https://www.uminho.pt/PT/uminho/Etica/comissoes-etica/Paginas/CEICSH.aspx>) da Comissão Especialização para a Educação nas Ciências Sociais e Humanas da Universidade do Minho (UM), no que se refere às normas orientadoras e procedimentos éticos para as ciências sociais e humanas da UM, bem como, os procedimentos a adotar na coleta de dados no domínio ético.

Este trabalho de investigação científica implicou princípios fundamentais, como o respeito pela dignidade da pessoa humana, pelo progresso e valorização do conhecimento, pela verdade científica e pela verdade de investigação.

4. APRESENTAÇÃO E ANÁLISE DE RESULTADOS

Este capítulo pretende apresentar os resultados obtidos no decorrer desta investigação, através da aplicação dos instrumentos de recolha de dados descritos no capítulo anterior. Proceder-se-á a uma análise dos resultados obtidos a partir da aplicação dos distintos instrumentos de recolha de dados, com uma descrição pormenorizada.

4.1. Teste de Pensamento Computacional

Como forma de avaliar os conhecimentos e competências dos alunos participantes na área das ciências da computação, mais concretamente no desenvolvimento das dimensões do Pensamento Computacional, foram aplicados dois testes: um pré-teste anterior ao estudo/intervenção e um pós-teste posterior ao estudo, sendo o enunciado completo dos mesmos disponibilizados pelo **(ANEXO H)**. Estes dois testes foram aplicados a mais de 120 alunos, mais especificamente, 138 alunos no pré-teste e a 125 alunos no pós-teste (tabela 10). Esta discrepância de 13 alunos deve-se, principalmente a situações de isolamento por motivo de COVID-19, fator impeditivo da participação dos alunos no pós-teste.

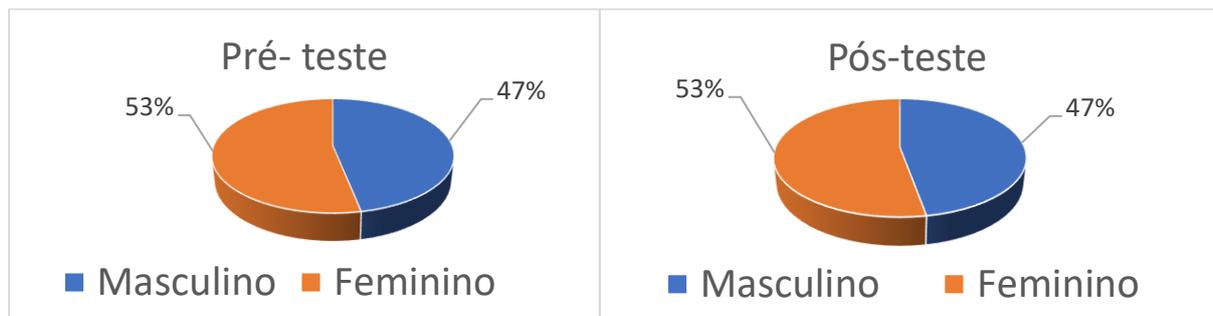
Tabela 10 - Distribuição de participantes por género e escolaridade

Escolaridade	Pré-teste			Pós-teste		
	Masculino	Feminino	Total	Masculino	Feminino	Total
3.º Ano	42	31	73	34	29	63
4.º Ano	28	37	65	25	37	62
	60	68	138	59	66	125

Fonte: Elaborado pelo investigador

Tal como se verifica na tabela anterior, a amostra é constituída por 138 alunos no pré-teste e 125 alunos no pós-teste, destes 73 alunos do 3.º ano realizaram o pré-teste e 63 alunos no pós-teste; já no 4.º ano 65 alunos realizaram o pré-teste e 62 o pós-teste. Como já foi referido anteriormente, a diferença de alunos que não realizaram o pré-teste, incidiram numa das turmas do 3.º ano, que à data do pós-teste se encontravam a recuperar do COVID-19.

Figura 27 - Índice do género da amostra



Fonte: Elaborado pelo investigador

Relativamente à distribuição dos participantes, apresentam uma distribuição homogénea de géneros, onde se verifica um equilíbrio entre ambos.

Os testes de PC foram aplicados e os dados foram recolhidos em março (pré-teste) e no final do ano letivo (pós-teste), mais concretamente em junho e, após a realização das avaliações finais do ano. Durante a aplicação do pré-teste e pós-teste, o investigador esteve sempre presente, acompanhando o desenvolvimento dos acontecimentos e auxiliando as professoras cooperantes de OCPR na sua concretização.

Os questionários usados neste estudo (pré-teste e pós-teste), são formados por questões de resposta fechada, descritos na secção 3.6.1. Para além das 28 questões, foram introduzidas, na parte final, algumas questões sobre a perceção de autoeficácia (numa escala tipo Likert, de “1=Péssimo até 5=Excelente”). Pretendeu-se, determinar o grau de satisfação dos alunos na realização do teste de Pensamento Computacional.

Bandura (1997), define autoeficácia como um julgamento que um determinado individuo faz sobre a sua capacidade de organizar e executar um conjunto de ações por forma a alcançar um determinado objetivo.

Figura 28 - Exemplo de uma questão sobre a perceção de autoeficácia

Q. Numa escala de 1 a 5, como achas que te correu o teste?
1 - Péssimo
5- Excelente

answer choices

- 1 - Péssimo (angry face emoji)
- 2 - (neutral face emoji)
- 3 - (neutral face emoji)
- 4 - (happy face emoji)
- 5 - Excelente (grinning face emoji)

Fonte: Elaborado pelo investigador

Deste modo, quando considerada no âmbito da educação, a autoeficácia representa a percepção do aluno da sua capacidade de organizar e concretizar determinados procedimentos que levam a gerar certos resultados relacionados com a sua aprendizagem (Bandura, 1993). Assim, será expectável considerar a autoeficácia mediadora do processo de desenvolvimento da aprendizagem.

4.1.1. Procedimentos de recolha

A definição dos procedimentos inerentes à recolha de dados está subordinada ao tipo de informação necessária ao esclarecimento dos objetivos de investigação. Assim, neste estudo, julgámos pertinente aplicar o Teste de Pensamento Computacional (pré-teste e pós-teste) como técnica, a fim de identificar os níveis de desenvolvimento do Pensamento Computacional (OBE3), em crianças do 1.º CEB.

Para coordenar a aplicação do Teste de Pensamento Computacional, foram realizados diversos procedimentos técnicos e logísticos, nomeadamente:

- Transposição para a plataforma digital (quizizz) das questões do Teste de Pensamento Computacional;
- Experimentação do teste com um grupo de alunos (Teste piloto);
- Solicitação de autorização às diferentes escolas envolvidas para utilização de equipamento informático (portáteis e *tablets*) e salas para o dia da aplicação do teste;
- Garantia de um computador/ *tablet* por aluno, com acesso à internet e em funcionamento;
- Garantia da velocidade de internet suficiente para a realização do teste;
- Ter em especial atenção ao modo, ao momento e às condições de aplicação do instrumento em meio escolar, devendo fazer-se em estreita colaboração com os professores participantes e a direção do agrupamento;
- Durante a aplicação, verificar os dados recolhidos;
- Previsão de situações que pudessem impossibilitar a aplicação do teste e propor soluções.

Conforme sugestão do investigador, em concordância com a direção das escolas participantes e professores cooperantes, a primeira recolha (pré-teste) ocorreu em março de 2022, na escola sede (Escola Secundária). No dia da aplicação do teste, os alunos deslocaram-se à escola sede, fazendo-se acompanhar pelos professores titulares de turma, professoras cooperantes e funcionários que auxiliam na deslocação entre as escolas (a distância entre as escolas é aproximadamente 1 km).

Todos os intervenientes/alunos estavam previamente informados sobre o dia e hora da aplicação do inquérito (pré-teste). Neste sentido, foi realizado um escalonamento com a indicação da turma, hora, sala TIC e duração. Para os alunos, o simples facto de se deslocarem a um espaço que não é o seu habitat natural terá criado um sentido de motivação extra e descontração, retirando alguma ansiedade que pudesse ter existido.

A aplicação teve a duração de cerca de 60 minutos e foi sempre realizada pelo investigador, com a coadjuvação de alguns alunos do 12.º ano e dos professores intervenientes, tomando as seguintes considerações, para garantir que a aplicação fosse, a mais idêntica possível, com todos os participantes e em todas as salas:

- Uma intervenção inicial com a turma para criar empatia, explicitando o objetivo do questionário e a importância do seu contributo nesta investigação, devendo, por isso ser respondido com sinceridade e empenho;
- Alerta para as regras ao longo da resolução: preenchimento de forma individual; não permissão de consulta nem apresentação de dúvidas ou tecer comentários ao longo da prova;
- Esclarecimento reforçado de que não existia um contributo direto para avaliação, não sendo obrigatório responder a todas as questões;
- Leitura e exploração conjunta dos exemplos relativos ao teste de pensamento computacional na parte inicial do inquérito;
- Informação de que, caso termine antes do tempo estipulado, o aluno deverá permanecer no local atribuído em silêncio.

No que concerne à segunda recolha (pós-teste), ocorreu em finais do mês de junho de 2022, após a realização das avaliações finais. Todas as considerações elencadas anteriormente para o pré-teste foram consideradas no pós-teste. No entanto, não foi possível aplicar o inquérito num só dia a todas as turmas, por questões de ocupação de espaços, na escola sede como concretizado no pré-teste.

Figura 29 - Aplicação do Teste de Pensamento Computacional (Pós-teste)



Fonte: Foto tirada pelo investigador

Nesta situação, a aplicação foi concretizada nas escolas associadas aos participantes e acompanhada pela professora cooperante, responsável pela leção da disciplina de OCPR e pelo investigador, de acordo com o horário da turma (tabela 1).

4.1.2. Dificuldades encontradas

Numa investigação, existem sempre variáveis que influenciam o estudo, criando desafios e dificuldades num processo dinâmico, no qual o investigador tem de se adaptar ao campo estudado, na medida que não existem desenhos universais, nem campos perfeitos. A investigação constrói-se sobre as características do investigador e do contexto (Vilelas, 2020).

No processo de aplicação dos questionários (pré-teste e pós-teste), destacam-se alguns constrangimentos na coordenação de horários, espaços físicos e logística, devido ao elevado número de alunos. Foi necessário coordenar junto da direção das escolas envolvidas a preparação do dia da aplicação do teste. No pré-teste, foi possível agrupar todos os alunos (138) numa tarde e concretizar a aplicações do teste, já no pós-teste foi excluída essa via por diversas razões, como a falta de funcionários para acompanhar os alunos à escola sede, condições atmosféricas adversas, realização de exames e indisponibilidade de salas TIC.

No decorrer do processo de recolha do pós-teste, surgiram também constrangimentos no âmbito do acesso à internet numa das escolas, sendo necessário o reagendamento para um dia alternativo. Estes contratempos na aplicação do inquérito (pós-teste), levaram a um prolongamento da aplicação do instrumento até ao final do mês de junho.

O investigador observou durante a aplicação do pós-teste, que os alunos estavam mais distraídos e inquietos comparativamente com o momento do pré-teste. De facto, os constrangimentos descritos anteriormente, prolongaram a aplicação do pós-teste para o final do mês de junho. Certamente, que as crianças não são máquinas e no início do verão, como é natural, alguns têm os seus irmãos de férias e eles ainda têm de ir à escola. Por outro lado, é no final do ano que as escolas organizam atividades lúdicas, procurando promover aprendizagens significativas em momentos de prazer, superação e bem-estar.

A adequação dos tempos, também no plano psicológico, é um fator que influencia a predisposição para as aprendizagens e no final do mês de junho, os alunos acusam um elevado nível de cansaço e dificilmente conseguem ter a indispensável capacidade de concentração num tempo que deveria ser de férias.

Face ao exposto anteriormente, será expectável que o comportamento das crianças tenha alguma influência nos resultados obtidos no teste de pensamento computacional (pós-teste), o que será objeto de análise no próximo tópico.

4.1.3. Resultados do teste de pensamento computacional

Nas próximas linhas, serão apresentados os resultados do estudo antes e após o programa de intervenção descrito na secção 3.5. Este estudo pretende analisar em que medida é que o programa desenvolvido, através de atividades de RE, contribui para melhorar o pensamento computacional dos alunos.

O estudo baseou-se numa intervenção empírica modelo quase-experimental, com avaliação de um pré-teste e de um pós-teste, num grupo de alunos de um determinado agrupamento.

Os resultados obtidos, encontram-se na sua totalidade no **(ANEXO I)**, numa escala de 0 a 28 pontos (1 ponto por cada resposta correta). Nessa tabela, está representado o número de questões (1 a 28), o número de respostas corretas e erradas por ano (3.º e 4.º) e por variação física (pré-teste e pós-teste). Inclui também uma representação, em termos percentuais, dos conceitos das ciências de computação referidos na tabela 12 e no estudo de Brackmann (2017) e González (2016), agrupados por questões (4 em 4), por ano e apresentado as variações totais.

Dessa tabela, foram retirados os dados mais importantes, culminando numa tabela mais sintetizada:

Tabela 11 - Resultados do Teste do Pensamento Computacional em termos percentuais

Questões		Pré-teste			Pós-teste			Comparação		
Conceitos		3.º ano	4.º ano	Total (3.º+4.º)	3.º ano	4.º ano	Total (3.º+4.º)	3.º ano	4.º ano	Total
Sequências simples	1	37,3 %	59,6%	47,83%	57,9%	58,9%	58,40%	20,6%	-0,7%	11%
	2									
	3									
	4									
Ciclos de repetição (<i>repeat x times</i>)	5	41,1%	55,0%	47,64%	49,6%	64,1%	56,80%	8,5%	9,1%	9%
	6									
	7									
	8									
Ciclos de repetição (<i>repeat until</i>)	9	38,0%	50,8%	44,02%	45,6%	50,8%	48,20%	7,6%	0,0%	4%
	10									
	11									
	12									
Condição simples (<i>if</i>)	13	26,4%	22,3%	24,46%	31,7%	32,3%	32,00%	5,4%	10,0%	8%
	14									
	15									
	16									
Condição composta (<i>if/else</i>)	17	36,0%	29,2%	32,79%	43,3%	40,3%	41,80%	7,3%	11,1%	9%
	18									
	19									
	20									
Ciclos de repetição (<i>while</i>)	21	28,8%	25,8%	27,36%	30,6%	32,3%	31,40%	1,8%	6,5%	4%
	22									
	23									
	24									
Funções simples	25	42,8%	44,2%	43,48%	49,6%	50,8%	50,20%	6,8%	6,6%	7%
	26									
	27									
	28									

Fonte: Criado pelo investigador

Tal como se pode verificar na tabela anterior, ao analisarmos os resultados obtidos pelos alunos neste estudo, mais especificamente ao compararmos os resultados obtidos pelos alunos na variação física antes e depois de ser aplicado o programa de intervenção, podemos considerar que, na sua globalidade, os resultados após apresentam um ganho considerável em termos percentuais, relativamente aos pré, havendo indicadores de que a intervenção poderá estar a ter um impacto positivo.

Neste seguimento, importa avaliar os impactos da intervenção nos conhecimentos dos alunos, em termos de conceitos fundamentais das Ciências de Computação, respondendo assim a um dos objetivos desta investigação (**OBJ3**), sendo necessário comparar os resultados obtidos nos 7 grupos de questões do teste do pensamento computacional (tabela 11).

No primeiro grupo, da questão 1 a 4, cujo objetivo passou por trabalhar as sequências simples, verifica-se que foram os alunos do 3.º ano que mais evidenciaram a evolução, o que poderá ser justificado por ser a primeira vez que têm contacto com a RE e conceitos de programação. Já os alunos do 4.º ano não evoluíram. Provavelmente estes alunos já tinham desenvolvido naturalmente parte destes conceitos no

ano anterior ou no seu dia a dia. Ainda que não seja possível provar, relembramos que foram os alunos do 3.º ano que evidenciaram um melhor resultado no concurso *Bebras* (secção 3.5.2).

No segundo grupo de questões, da questão 5 à 6, na utilização de ciclos de repetição (*repeat x times*), ambos os grupos demonstraram adquirir competências e entendimento do conceito. Ainda nas estruturas de repetição, com a utilização do ciclo de repetição até uma determinada condição, presente no terceiro grupo, da questão 9 à 12, o grupo de alunos do 3.º ano foi aquele que expressou uma melhoria significativa. Já no que concerne a utilização do ciclo de repetição enquanto uma condição estiver presente, sexto grupo, da questão 21 à 23, verifica-se uma melhoria nos dois grupos, mais acentuada, nos alunos do 4.º ano.

Em suma, de acordo com os resultados obtidos, tudo indica que o conceito de estruturas de repetição, que são estruturas que permitem a execução de uma ou mais instruções repetidamente até que uma condição, também conhecidas por (*loops*), foram de acordo com a análise dos dados obtidos, conceitos interiorizados pelos alunos, ainda com algumas dificuldades. Contudo, a intervenção realizada nas aulas de OCPR, fornece indicadores positivos na aquisição de conceitos de ciências de programação e nas dimensões do Pensamento Computacional (Abstração, Decomposição, Reconhecimento de Padrões, Algoritmos e Depuração).

Nos grupos de questões relacionadas com condições simples e compostas, da 13 a 20, ambos os grupos evidenciaram resultados positivos, tornando-se visível que os alunos do 4.º ano obtiveram uma percentagem de aquisição de conhecimentos acima dos 10% após a intervenção das atividades com RE.

As funções simples não foram objeto de exercícios específicos no âmbito da intervenção, mas os conceitos foram abordados de uma forma superficial em algumas atividades, principalmente atividades que requereram o uso da placa Micro:Bit. Os resultados das questões 25 à 28 foram idênticos em ambos os grupos o que denota que conhecem o conceito e a aplicabilidade prática, mas não exercitaram o suficiente de forma a elevarem o conhecimento desta temática.

Considerando apenas a diferenças percentuais obtidas, pode afirmar-se que os alunos após o programa de intervenção apresentam, de uma forma geral, um incremento considerável do pensamento computacional, no entanto, estes resultados carecem de testes complementares para determinar a sua relevância estatística.

Neste seguimento, com recurso do *software* IBM SPSS – Statistic Versão 28, analisou-se os dados obtidos.

Tabela 12 - Resultados do Teste do Pensamento Computacional

Conceitos	Questões	Pré-teste			Pós-teste		
		N	Desvio Padrão	Média	N	Desvio Padrão	Média
Sequências simples	1	138	0,293	0,91	125	0,215	0,95
	2	138	0,470	0,33	125	0,502	0,49
	3	138	0,497	0,43	125	0,495	0,58
	4	138	0,437	0,25	125	0,465	0,31
Ciclos de repetição (<i>repeat x times</i>)	5	138	0,470	0,67	125	0,438	0,74
	6	138	0,473	0,67	125	0,424	0,77
	7	138	0,452	0,28	125	0,482	0,36
	8	138	0,452	0,28	125	0,492	0,40
Ciclos de repetição (<i>repeat until</i>)	9	138	0,490	0,61	125	0,458	0,70
	10	138	0,502	0,51	125	0,502	0,52
	11	138	0,488	0,38	125	0,497	0,43
	12	138	0,437	0,25	125	0,447	0,27
Condição simples (<i>if</i>)	13	138	0,498	0,44	125	0,501	0,53
	14	138	0,380	0,17	125	0,471	0,33
	15	138	0,398	0,20	125	0,434	0,25
	16	138	0,374	0,17	125	0,382	0,18
Condição composta (<i>if/else</i>)	17	138	0,468	0,32	125	0,477	0,34
	18	138	0,452	0,28	125	0,495	0,42
	19	138	0,494	0,41	125	0,502	0,52
	20	138	0,459	0,30	125	0,490	0,39
Ciclos de repetição (<i>while</i>)	21	138	0,494	0,41	125	0,500	0,46
	22	138	0,380	0,17	125	0,382	0,18
	23	138	0,338	0,13	125	0,284	0,09
	24	138	0,486	0,38	125	0,501	0,54
Funções simples	25	138	0,459	0,30	125	0,486	0,38
	26	138	0,499	0,45	125	0,502	0,51
	27	138	0,490	0,39	125	0,501	0,47
	28	138	0,491	0,60	125	0,480	0,65

Fonte: Criado pelo investigador, com dados retirados do *software* IBM SPSS – *Statistic* Versão 28

Na tabela anterior, encontram-se as médias obtidas, em cada questão, pelos alunos dos 3.º e 4.º anos que frequentam as aulas de OCPR, antes e depois da intervenção com atividades de RE. Os resultados traduzem-se obviamente por uma igualdade com os já apresentados anteriormente no formato de estatística simples (Tabela 11).

Ao analisarmos os dados alcançados neste estudo, mais concretamente ao compararmos os resultados obtidos pelos alunos antes e depois da intervenção de atividades com RE, pretendemos verificar se há diferenças de médias no domínio do pensamento computacional.

Tabela 13 - Médias pré-teste e pós-teste

Momento	Total de alunos	Média	Diferença
Pré-teste	138	10,70	2,16
Pós-teste	125	12,86	

Fonte: Criado pelo investigador, com dados retirados do *software* IBM SPSS – *Statistic* Versão 28

Considerando apenas a diferença entre as médias obtidas, verifica-se uma ligeira diferença de incremento após a intervenção.

Antes de avançar, será necessário realizar um teste de normalidade, de forma a aferir se as amostras se enquadram numa distribuição normal ou não.

Tabela 14 - Resultado dos Testes de Normalidade

Testes de Normalidade						
Momento	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estatística	gl	Sig.	Estatística	Gl	Sig.
Pré-teste	,077	138	,046	,976	138	,016
Pós-teste	,095	125	,008	,973	125	,014

a. Correlação de Significância de Lilliefors

Fonte: Criado pelo investigador, com dados retirados do *software* IBM SPSS – *Statistic* Versão 28

Analisando a tabela anterior e considerando os valores apresentados pelos testes de normalidade (Kolmogorov-Smirnov e Shapiro-Wilk), é evidente que valores de *p-value*, se encontram abaixo ou muito próximos de 0,05 no nível de significância (pré-teste, K-S *p-value* = 0,046 (4,6 %) e S-W *p-value* = 0,016 (1,6 %), pós-teste, K-S *p-value* = 0,008 (0,8 %) e S-W *p-value* = 0,014 (1,4 %)).

Os investigadores Lopes et al.(2013), consideram que o teste de Shapiro-Wilk é o mais indicado para amostras de pequena dimensão e o teste de Kolmogorov-Smirnov, com a correção de Lilliefors, para amostras de grandes dimensões ($n \geq 30$). A regra de decisão para estes testes de ajustamento consiste em não rejeitar H_0 quando *p-value* $\geq \alpha$, sendo α o nível de significância do teste. No entanto, alguns autores, defendem que o teste de Shapiro-Wilk é o mais indicado, mesmo em amostras de menor dimensão (Razali & Wah, 2011; Torman et al., 2012).

No presente estudo, considerando a necessidade de compararmos os resultados obtidos pelos alunos na variação física antes (pré-teste) e depois (pós-teste), a amostra de alunos considerada foi a do pós-teste (125 alunos), com o objetivo de obtermos resultados de duas amostras emparelhadas. Deste modo, procedemos a uma nova avaliação de normalidade para uma amostra de N=125, para os mesmos alunos antes e depois da intervenção.

Tabela 15 - Resultado dos Testes de Normalidade do pré-teste e do pós-teste

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estatística	gl	Sig.	Estatística	gl	Sig.
Pré-teste	,085	125	,026	,971	125	,009
Pós-teste	,095	125	,008	,973	125	,014

a. Correlação de Significância de Lilliefors

Fonte: Criado pelo investigador, com dados retirados do software IBM SPSS – Statistic Versão 28

A utilização deste teste pressupõe a definição de duas hipóteses:

H_0 = A distribuição dos dados é normal ($p > 0,05$), ou seja, 5%;

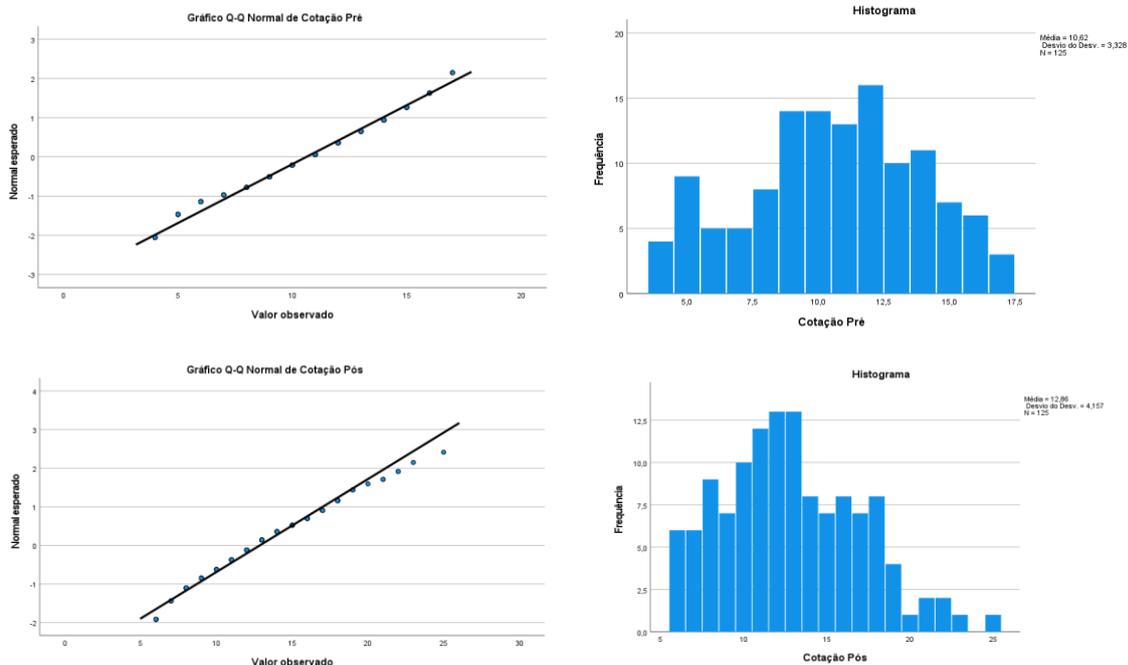
H_1 = A distribuição dos dados é diferente de normal ($p < 0,05$).

Ao analisar a tabela anterior e tendo em conta os dois testes, o de Kolmogorov-Smirnov e Shapiro-Wilk, para um nível de significância (α) de 5%, fica validado o pressuposto que as variáveis em todas as amostras provêm de uma população não normal, ou seja, não-paramétrica, isto porque os resultados obtidos no nível de significância são menores que 5%, rejeitando assim a hipótese (H_0).

pré-teste, K-S p -value = 0,026 (2,6 %) e S-W p -value = 0,009 (0,9 %)

pós-teste, K-S p -value = 0,008 (0,8 %) e S-W p -value = 0,014 (1,4 %)

Gráfico 1 - Gráficos de normalidade do pré-teste e do pós-teste



Fonte: Criado pelo investigador, com dados retirados do software IBM SPSS – Statistic Versão 28

Nos gráficos anteriores, pode observar-se a aleatoriedade dos pontos relativamente à reta e a distribuição normal.

Existindo evidência estatística suficiente para afirmar que os resultados das respostas corretas ao teste de pensamento computacional, antes e depois da intervenção, se afastam da distribuição normal, pelo que se assume que os valores obtidos não seguem uma distribuição normal.

Tendo em consideração os resultados da normalidade da amostra, estamos em condições de aplicar testes não-paramétricos, nomeadamente o teste de Wilcoxon, uma vez que permite fazer a análise da diferença entre duas medianas, num contexto de duas amostras emparelhadas. Este teste permite comparar a mesma variável dependente (número de respostas corretas do Teste de Pensamento Computacional) em dois momentos temporais (pré-teste e pós-teste), que se adequa perfeitamente a esta investigação.

De forma a explorar os resultados, foram equacionadas duas hipóteses: a primeira, a hipótese nula, em que se verifica a ausência de diferenças entre as medianas dos grupos; e a segunda onde se verifica que as medianas são diferentes, ou seja, as hipóteses são:

H_0 : Não há diferença entre os resultados do pré-teste do PC e o pós-teste do PC

H_1 : Há diferença entre os resultados do pré-teste do PC e o pós-teste do PC

Tabela 16 - Ordenações, resultado do teste de Wilcoxon

		Postos		
		N	Posto médio	Soma de Classificações
Cotação Pós - Cotação Pré	Classificações Negativas	9 ^a	35,44	319,00
	Classificações Positivas	101 ^b	57,29	5786,00
	Empates	15 ^c		
	Total	125		

a. Cotação Pós < Cotação Pré

b. Cotação Pós > Cotação Pré

c. Cotação Pós = Cotação Pré

Fonte: Criado pelo investigador, com dados retirados do *software* IBM SPSS – *Statistic* Versão 28

A tabela anterior representa os resultados da amostra com as respetivas ordenações/ordem, destaca-se o facto de 9 alunos tiraram um resultado pior no pós-teste, comparativamente como o pré-teste. Já 101 alunos evidenciaram um resultado superior no pós-teste e 15 alunos mantiveram a classificação em ambos os testes.

Gráfico 1 - Teste Wilcoxon – Informações de ordem

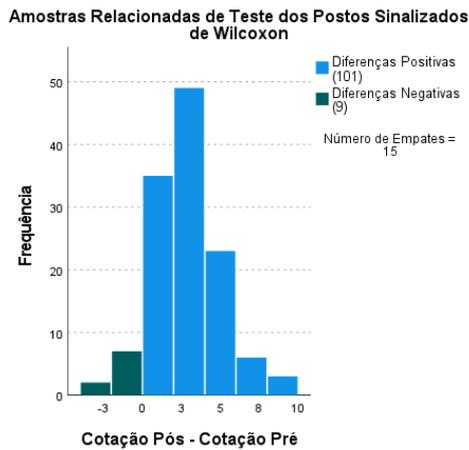


Gráfico 2 - Teste Wilcoxon – Informações Grupo Pré

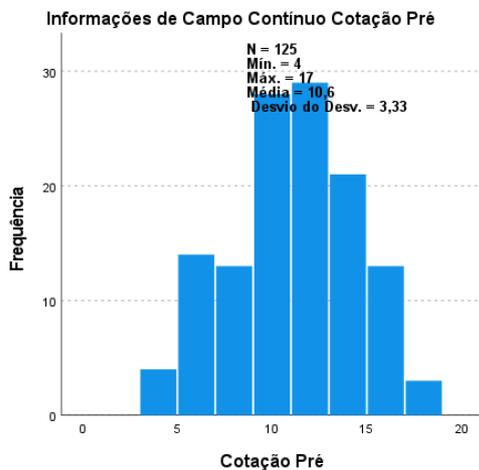
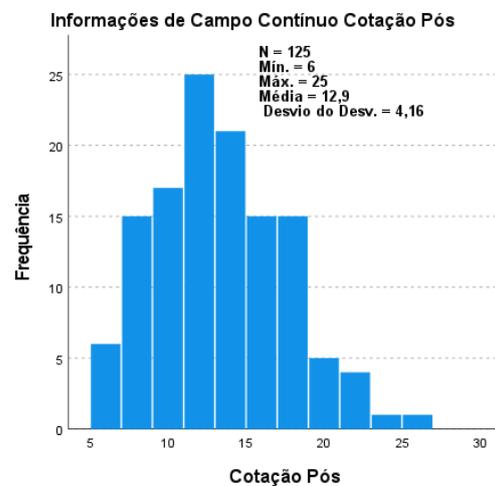


Gráfico 3 - Teste Wilcoxon – Informações Grupo Pós



Fonte: Criado pelo investigador, com dados retirados do software IBM SPSS – Statistic Versão 28

Do *output* do SPSS é apresentada uma outra tabela, a tabela de estatística que informa o valor de teste de Wilcoxon.

Tabela 17 - Resultado do teste de Wilcoxon - Estatísticas

Estatísticas de teste^a

Cotação Pós - Cotação Pré	
Z	-8,207 ^b
Significância Sig. (2 extremidades)	<,001

a. Teste de Classificações Assinadas por Wilcoxon

b. Com base em postos negativos.

Fonte: Criado pelo investigador, com dados retirados do software IBM SPSS – Statistic Versão 28

Ao analisarmos a tabela anterior, aparecem dois valores, o valor do Teste de Wilcoxon, representado por Z e o respetivo valor p (significância), nomeadamente $Z = -8,207$ e $p\text{-value} < 0,01$.

Com base nos resultados obtidos, podemos evidenciar que o $p\text{-value} < 0,01$, portanto inferior a 5%, rejeitando a hipótese nula (H_0), pelo que se deve concluir que há diferença entre os valores obtidos entre os resultados do pré-teste do PC e o pós-teste do PC. Existe assim uma evidência estatística suficiente para afirmar que os alunos, após uma intervenção com atividades de Robótica Educativa evidenciam um aumento significativo do conhecimento relacionado com as dimensões do Pensamento Computacional.

Os resultados obtidos, e após a análise da tabela com estatística simples (tabela 11) e análise estatística descrita anteriormente, na sua generalidade, são semelhantes. Importa salientar que os resultados vão ao encontro do padrão de resultados obtidos em outros estudos Brackmann(2017), Marques (2021) e Román-González et al. (2018, 2019), destacando-se que existe uma tendência para um contributo positivo em relação ao Pensamento Computacional, quando são aplicados programas/atividades com cariz semelhante à intervenção aplicada neste estudo, nomeadamente relacionados com a Robótica Educativa.

4.2. *Focus group* professores

No dia 07 de julho de 2022, pelas 11:00, na sala anexa à biblioteca da escola sede, procedeu-se ao início da sessão de *focus group*, agradecendo a participação de todas as professoras (professoras titulares de turma e professoras cooperantes de OCPR do 3.º e 4.º ano), referindo-se à necessidade de se proceder à gravação da sessão (apenas áudio), de forma a ser transcrita e analisada, garantindo o anonimato e solicitando a autorização dos presentes. Foi autorizada por unanimidade.

A realização deste *focus group* teve como pretensão contribuir para as respostas aos objetivos de investigação, nomeadamente ao objetivo 1 – descrever metodologias a implementar, ao objetivo 2 – caracterizar materiais pedagógicos que suportem as atividades e ao objetivo 3 – identificar os níveis de desenvolvimento do Pensamento Computacional.

O investigador iniciou o *focus group* cumprimentando e agradecendo a presença dos professores. De seguida, informou os presentes sobre o tema, que está relacionado com as atividades de Robótica Educativa e com o desenvolvimento do Pensamento Computacional, no âmbito da disciplina de OCPR. Referiu ainda, que o *focus group*, é uma conversa onde os intervenientes poderão relatar e partilhar as experiências e opiniões sobre a metodologia aplicada, no âmbito da disciplina de OCPR, nomeadamente: na importância da RE e da programação, na aprendizagem dos alunos e no nível de consolidação de

conhecimentos; de que forma, a RE promoveu as dimensões PC; qual é avaliação da metodologia aplicada nas aulas de OCPR e como caracterizam os materiais de RE adotados durante as aulas de OCPR.

4.2.1. Apresentação de Resultados

Por forma a clarificar o conteúdo da entrevista por *focus group*, foi realizado o seguinte quadro-síntese:

Tabela 18 - Quadro Síntese Focus Group

Categoria / Tema / Abordagem	Evidências /Respostas
Materiais 1.º Ciclo - RE	<p>Existe muito equipamento (robôs) e condições. Fez-se algum investimento. Não há tempo para explorar todo o equipamento. Precisa de melhorar as infraestruturas (exemplos: melhorar a rede <i>Wireless</i> e sistema de climatização).</p>
Competências desenvolvidas - aulas OCPR	<p>Os alunos gostam muito pelo espírito colaborativo, liderança, trabalho em grupo, respeito pelos colegas, orientação, escuta ativa, desenvolvem a atenção e a imaginação e certos conceitos. Raciocínio lógico concreto ex. Matemática.</p>
Metodologias a implementar – RE	<p>Os projetos fora da escola significam necessidade de mais tempo - exige mais envolvimento e motivação pessoal por parte dos professores. O processo é mais envolvente, com toda a turma. Atividades conjuntas e com boa articulação. Colaboração dos alunos do profissional. Desenvolvimento de aulas mais práticas.</p>
Transversalidade RE com outras áreas curriculares	<p>Totalmente verdade. A transversalidade é crucial na articulação com outras áreas curriculares. Existem conceitos que se pode utilizar e desenvolver na robótica.</p>
Importância RE no 1.º ciclo	<p>Desenvolver o raciocínio, o trabalho de grupo, a autonomia, o pensamento crítico, o desenvolvimento de novas ideias.</p>
Importância uso robôs e programação no 1.º ciclo	<p>Compreendem melhor a adequação das tecnologias no dia-a-dia. Desenvolvem a criatividade para criação de outros robôs no futuro. Verificam a importância da programação. Estimulam o desenvolvimento.</p>
Aspetos mais relevantes nas aulas de OCPR da reação dos alunos	<p>Perceção das suas capacidades e competências. Articulação de conhecimentos de várias disciplinas. Maior conhecimento sobre programação e Pensamento Computacional. Desenvolvimento de atividades através do raciocínio.</p>
Contributo dos professores 1.º ciclo nas aulas OCPR	<p>Sugestões dadas pelos professores. A articulação das diferentes disciplinas.</p>

	Outra percepção diferente dos professores de robótica que complementa a opinião do professor titular.
Projetos mais relevantes em robótica, que se refletiram na aprendizagem dos alunos nas diferentes disciplinas	Destaque do projeto em formato teatro (pelo envolvimento de todas as disciplinas). Projetos relacionados com a Matemática e o inglês.
Alterações propostas no ensino da OCPR	Horário Tempo Condições técnicas Espaço Cada aluno ter o seu computador
Projeto da RE: continuidade? Sim ou Não	Sim Todas as vantagens e mais-valias abordadas, realçam a importância da continuidade do projeto.
Avaliação da OCPR neste ano letivo	“Bastante Positiva”

Fonte: Quadro elaborado pelo investigador, com informações retiradas da gravação do *focus group*

4.2.2. Análise dos resultados do *focus group* de professores

Após a análise e estudo da entrevista realizada aos professores do 1.º ciclo, constata-se que de uma forma abrangente, as atividades desenvolvidas na disciplina de OCPR foram **muito enriquecedoras**, pelo desenvolvimento dos alunos e pela **articulação** com os professores das diferentes disciplinas. Permitiu abordar os **conceitos científicos** interligando-os com a prática.

Relativamente aos materiais, mencionaram com agrado que o agrupamento dispõe de uma grande diversidade de **conjuntos robóticos**, no entanto, a infraestrutura de rede nas escolas do 1.º ciclo é limitada. “É impossível, de todo, utilizar a rede para uma turma”, foi a maior dificuldade encontrada.

Foi também referido que “é importante planificar atividades para aplicar o equipamento, mas para isso é preciso haver ideias, **tempo** e todos trabalharem de forma colaborativa”. Planificar, mas não de forma rígida, com objetivo de desenvolver o raciocínio na resolução de problemas e a lógica na construção de artefactos robóticos, bem como desenvolver métodos de trabalho de forma colaborativa.

Verificaram-se algumas condicionantes devido à pandemia, muitas oscilações e alterações ao longo do ano letivo, apesar de existirem menos problemas do que no ano anterior.

Foi dito que “os miúdos de hoje são muito egocêntricos e não ouvem e, nesta disciplina OCPR desenvolvem o **espírito colaborativo**, escuta ativa e respeito pelo outro.”

Temos o exemplo de alunos que desenvolvem menos competências em outras disciplinas e em OCPR conseguiram um desenvolvimento maior em competência de **raciocínio lógico**.

A robótica contribui com muitas **competências**, mas 1h por semana é muito pouco para desenvolver outras aptidões, como é o caso da **resolução de problemas**, a criatividade, a autonomia, a tomada de decisão. Para isso é necessário um trabalho a longo prazo.

“Ajuda muito na concentração e, sendo um trabalho de grupo, faz com que o respeito pela opinião do outro seja cultivada, o que é muito importante.”

A disciplina de OCPR, existe desde 2016, através da qual são desenvolvidas algumas atividades de robótica e de pensamento computacional. No entanto, apesar de algumas atividades com RE promoverem as dimensões do pensamento computacional, como a abstração, a decomposição e a algoritmia, são necessárias mais horas e mais tempo e um trabalho mais amplo.

Os alunos gostaram do projeto realizado este ano, deram como exemplo os jogos, a resolução de problemas, a aula de eletricidade. Tiraram proveito próprio através de um trabalho mais prático e com mais **articulação entre conceitos**, disciplinas e aplicabilidade.

Realçaram o exemplo de uma aluna que fez a Prova de Aptidão Profissional (PAP) e, todas as 5.ª feiras participava e ajudava os alunos mais novos a compreenderem algumas dinâmicas e conteúdos. Os alunos tinham os equipamentos, mexiam nos materiais, montavam, desmontavam e testavam. Permitiam o desenvolvimento de **competências técnicas** e de métodos de trabalho através do planeamento e envolvimento em projetos de robótica, de forma **colaborativa**.

Exemplificaram conteúdos e conceitos da Matemática para desenvolver a memória visual dos alunos. Na aula de robótica, conseguem memorizar melhor os conceitos através da prática visual.

As atividades realizadas no âmbito da disciplina de OCPR foram desenvolvidas em articulação com os professores titulares das turmas envolvidas. Esta mais-valia destacou-se tanto para os alunos como para os professores.

As atividades desenvolvidas com recurso à robótica educativa permitiram que os alunos aprendessem conceitos da área das ciências da computação e desenvolvessem a capacidade de programar. Estas atividades, permitiram contribuir para o **desenvolvimento dos princípios** do pensamento computacional nos alunos.

As atividades que os alunos mais gostaram de desenvolver nas aulas de OCPR foram: o teatro, a dança, o percurso da linha, montar e desmontar, teste de estudo do meio, aula de eletricidade e os jogos (exemplo: Micro:Bit).

Alguns alunos evidenciaram o facto de ficarem com a percepção muito positiva das suas capacidades nesta disciplina comparativamente com outras disciplinas e com os melhores alunos. Este facto evidencia que existem alunos com mais apetências para esta disciplina do que para outras.

Na implementação da robótica como uma **metodologia ativa**, foram realçados alguns pontos fortes: a possibilidade de passar do abstrato para o concreto, o desenvolvimento do raciocínio através da robótica, da programação, do Inglês e a resolução de problemas e desenvolvimento computacional. Como pontos fracos, destacou-se novamente o tempo e a infraestrutura de rede.

“O robô é um brinquedo, que desenvolve o raciocínio e leva a que os alunos queiram comprar robôs para desenvolver diferentes brincadeiras, o que lhes permite desenvolver mais a criatividade.”

Abordaram o facto de os alunos desenvolverem mais gosto por esta temática do que as alunas, apesar de considerarem que elas têm mais apetências para esta área do que os alunos. No entanto, consideram que os robôs e a manipulação dos mesmos, a programação e a montagem, estimulam a **motricidade fina**, fator extremamente importante no desenvolvimento da aprendizagem, sejam alunos ou alunas.

“A Robótica é, sem dúvida alguma, uma mais-valia no primeiro ciclo. Só a alegria das crianças nestas aulas merece logo uma avaliação muito positiva. A vontade que eles sentem em descobrir mais, saber como se faz, partilhar e sentir que são capazes, o entusiasmo com que desenvolvem os trabalhos, o raciocínio e o pensamento computacional.”

Quanto ao projeto Programação e Robótica, por todas as vantagens e mais-valias abordadas ao longo da entrevista *focus group*, os professores entrevistados, por unanimidade, realçaram a importância da continuidade deste projeto, fazendo uma avaliação “bastante positiva”, sem dúvida um projeto para continuar.

Na parte final do *focus group* o investigador dirigiu algumas questões específicas às professoras cooperantes que lecionaram a disciplina de OCPR, com o objetivo de explorar as percepções e experiências relevantes para o tema em estudo.

Na perspetiva das professoras cooperantes, de acordo com a metodologia aplicada, as atividades desenvolvidas com recurso à RE permitiram aos alunos desenvolver conceitos iniciais das ciências de

computação, nomeadamente no reconhecimento, exploração e utilização de estruturas de programação (Operadores Aritméticos, Relacionais e Lógicos, Variáveis e Constantes, Estruturas Condicionais e Ciclos).

Consideraram, de forma unânime, que os alunos revelaram alguma dificuldade na implementação de algumas estruturas condicionais e ciclos de repetição. No entanto, alguns alunos demonstraram aptidão para implementar uma sequência lógica de resolução de um determinado problema, com base nos fundamentos associados à lógica da programação, recorrendo a componentes estruturais da programação. Foram capazes de analisar e refletir sobre a solução encontrada e a sua aplicabilidade e reconheceram a necessidade de reformular a sequência lógica de resolução do problema, de forma colaborativa.

Relativamente às atividades implementadas em sala de aula através dos conjuntos robóticos (Lego WeDo 2.0, Micro:Bit, Robô Ozobot e alguns componentes eletrônicos), reconhecem que as atividades contribuíram para o desenvolvimento dos alunos ao nível da Robótica Educativa, nomeadamente na análise e compreensão do funcionamento de diversos mecanismos físicos e no desenvolvimento de métodos e organização do trabalho através do planeamento e envolvimento em projetos de robótica de forma colaborativa.

Com a RE os alunos construíram pequenas maquetes/cenários, através de elementos dinâmicos (engrenagens, motores, sensores), utilizando uma linguagem de programação visual para interagir com o artefacto robótico, permitindo desenvolver o raciocínio na resolução de problemas.

No que concerne ao desenvolvimento de conhecimentos adquiridos pelos alunos, no âmbito do Pensamento Computacional, as professoras participantes, referem que o nível de consolidação de conhecimentos é francamente positivo. Realçam a importância das atividades desenvolvidas na estimulação da curiosidade pela investigação, reconhecendo que os alunos desenvolveram competências do pensamento computacional, como a análise de problemas concretos, recorrendo a abstração e decomposição, para que possam ser modelados e tratados de forma independente. Ao longo das atividades, os alunos reconheceram, a partir de necessidades concretas, padrões unificadores (por exemplo no uso de ciclos) e criaram sequências de passos elementares, perfeitamente definidos, que permitiram criar as ações pretendidas. No final das atividades, foram capazes de comparar os resultados reais com os resultados pretendidos, bem como analisar e testar passo-a-passo as propostas de solução para um determinado problema, identificar possíveis erros e propor soluções mais eficazes.

5. REFLEXÃO GLOBAL SOBRE OS RESULTADOS

Com o culminar desta investigação e após a apresentação do enquadramento teórico, da metodologia e dos resultados, impõe-se uma reflexão/especulação global sobre os resultados que, no fundo, devem estar associados à procura das respostas aos objetivos desta investigação, nomeadamente em relação ao seu contributo para o desenvolvimento do pensamento computacional, nos alunos do 1.º ciclo.

Após comparação dos resultados obtidos nesta investigação através da aplicação do Teste de Pensamento Computacional de Román-González (*Computational Thinking Test – CTT*) aos alunos dos 3.º e 4.º anos do 1.º Ciclo e do *focus group*, com autores que estudam esta temática, apresenta-se a síntese dos resultados encontrados.

Relativamente ao **primeiro objetivo** – Descrever metodologias a implementar - as professoras cooperantes, que lecionam a disciplina de OCPR, consideram que a metodologia aplicada neste estudo permitiu aos alunos desenvolverem conceitos iniciais das ciências da computação e habilidades de PC. Através da literatura é evidente uma preocupação generalizada por um alargamento e um aprofundamento do conhecimento e das capacidades das crianças e jovens no domínio das tecnologias, reforçando a componente da computação e em particular a programação/codificação, até agora com pouca expressão curricular. Acentua-se a necessidade, para o desenvolvimento das crianças e jovens, em que “nada é mais importante do que aprender a pensar de forma criativa, aprender a encontrar soluções inovadoras para as situações inesperadas que certamente surgirão ao longo das suas vidas” sustenta Resnick (2007).

A metodologia é um dos aspetos mais importantes desta investigação, pois o uso RE como estratégia para disseminar ou promover os conceitos relacionados com o PC, requer uma planificação pedagógica ancorada em conceitos bem definidos em ciências da computação. É importante ressaltar que os conjuntos de robótica utilizados nesta investigação não se resumem meramente à utilização destes materiais, existindo o cuidado de planificar cada atividade proposta com um objetivo específico assente nas linhas orientadoras do Ministério da Educação, "Programação e Robótica no Ensino Básico - Probótica", nas propostas educativas “K-12 Computer Science Framework”, “K-12 Computer Science Standards” e “ISTE Computational Thinking Competencies” com a adaptação de acordo com a nossa realidade educativa e em atividades propostas por outros estudos no mesmo campo de investigação.

Caso contrário, a utilização desorientada de recursos pode desmotivar o aluno e comprometer a sua aprendizagem. É necessário utilizar teorias de aprendizagem para fundamentar a utilização da RE em sala de aula (Bakala et al., 2021; Guedes et al., 2021; Neto & Bertagnolli, 2021).

A abordagem UMC descrita na secção 2.3 foi utilizada nas diversas atividades, por fornecer suportes adicionais aos alunos com pouca ou nenhuma experiência em programação, um conceito de três estágios que ajuda a reduzir a ansiedade na programação e sustenta o crescimento do conhecimento das crianças e jovens (Lee et al., 2011). Esta metodologia baseia-se no princípio e na lógica de que as interações devem ser aprofundadas sucessivamente com suporte apropriado (conceito de *scaffolding*). Diversos estudos (Franklin et al., 2020; Körber et al., 2021; Lee et al., 2011; Rachmatullah et al., 2021), referem que a abordagem UMC melhora a aquisição das competências ao nível das ciências da computação e do PC e, ao mesmo tempo, diminui a ansiedade dos alunos e dos próprios professores relativamente à aprendizagem, fornecendo bases sólidas e um percurso linear no processo de aquisição de conhecimento. Em suma, pelo *feedback* de alunos e professores e pelas experiências evidenciadas pelos autores anteriores a progressão UMC pode ser um recurso metodológico na introdução de conceitos das ciências da computação e do PC, através da RE na escolaridade inicial. Vai ao encontro do construcionismo de Papert, de aprender por exploração, de ajustes e do tangível, porque no estágio de modificação envolve uma aprendizagem por experimentação e descoberta na manipulação do código ou artefacto.

No que se refere ao objetivo dois (tabela 2, da secção 3.2), **caracterizar materiais pedagógicos que suportam as atividades**, os professores relatam que os alunos ao utilizarem as diversas plataformas de programação visual no manuseamento dos três conjuntos robóticos (Lego WeDo 2.0, Placa Micro:Bit e Ozobot) utilizados nesta investigação potenciam a aprendizagem dos alunos na medida em que a utilização da robótica permite às crianças aprenderem a planear, a construir e programar, através da resolução sucessiva de problemas, para atingirem o objetivo final. Foi possível observar que ao manipularem estes instrumentos pedagógicos demonstraram um grande entusiasmo, interesse e empenho na concretização das atividades propostas. A utilização da RE permite às crianças desenvolver múltiplas e diversas áreas, tais como “o pensamento computacional, a algoritmia, a programação e ainda os robôs e outros objetos tangíveis programáveis” (Pedro et al., 2017, p.9). O mesmo autor salienta a importância da utilização da robótica por parte das crianças, desde tenra idade, na medida em que, devido à evolução da sociedade, as competências exigidas no século XXI, nomeadamente a “resolução

No que concerne à **identificação dos níveis de desenvolvimento do Pensamento Computacional**, os professores realçam a importância das atividades desenvolvidas, na estimulação da curiosidade pela investigação, reconhecendo que os alunos desenvolveram habilidades do pensamento computacional, como a análise de problemas concretos, recorrendo a abstração e decomposição, para que possam ser modelados e tratados de forma independente. Consideram que as atividades desenvolvidas com a RE permitiram desenvolver determinados conceitos básicos das ciências da computação e de dimensões do PC. De facto, deveremos referir que os resultados desta investigação encontram concordância com os resultados de alguns estudos que apontam para a utilização da robótica educativa como uma estratégia pedagógica potenciadora da aprendizagem de conceitos iniciais de programação e do desenvolvimento de competências de pensamento computacional (Bakala et al., 2021; Brackmann, 2017; Chalmers, 2018; Garneli et al., 2015; Guedes et al., 2021; Marques, 2021; Žáček et al., 2019). No entanto, vários estudos (J. Ramos & Espadeiro, 2014a; Relkin & Bers, 2021; Román-González et al., 2019) consideram que a avaliação do PC é importante para documentar o progresso da aprendizagem, aferir a eficácia das aulas, contribuir para o desenvolvimento do currículo e para a identificação de alunos com dificuldades de aprendizagem.

Sabemos que um construto não é sólido ou empiricamente produtivo até que se torne uma variável que possa ser medida e avaliada. A medição e avaliação do pensamento computacional foi desde sempre uma preocupação e um objetivo desta investigação. Neste momento, já existem técnicas e instrumentos de avaliação das aprendizagens em ambientes computacionais, tanto de um ponto de vista transversal (por exemplo, testes, escalas, grelhas de avaliação, portfólios, ferramentas de análise de projetos de programação automática como o Dr. *Scratch*, etc.), com estudos que demonstram validades desses instrumentos (Ramos, 2015; Román-González et al., 2019). Do ponto de vista longitudinal, cada vez mais surgem testes confiáveis que permitem estimar o nível de desenvolvimento do pensamento computacional dos alunos, desde do ensino pré-escolar ao final da escolaridade obrigatória (Relkin & Bers, 2021; Román-González et al., 2017; Zapata-Caceres et al., 2020), onde se inclui o teste de pensamento computacional (CTt) de Román-González, utilizado nesta investigação, onde os resultados obtidos evidenciam que existe uma tendência para um contributo positivo em relação ao Pensamento Computacional, quando são aplicados programas/atividades com cariz semelhante à intervenção aplicada neste estudo, nomeadamente relacionados com a Robótica Educativa.

6. CONCLUSÕES

Este capítulo apresenta as conclusões do trabalho desenvolvido. Faz-se referência às considerações finais da investigação, evidenciando as principais contribuições para a comunidade científica. Serão elencados alguns constrangimentos fruto de um trabalho empírico exposto a variáveis muitas vezes difíceis de controlar. Por fim, destacam-se algumas propostas de trabalho futuro.

Na última década, assistimos à multiplicação de iniciativas e programas de introdução do pensamento computacional na escola em diversos países, através do recurso a diferentes tecnologias, artefactos, linguagens de programação e ambientes computacionais.

A Robótica Educativa é considerada uma das tecnologias que mais impacto cria nas crianças e jovens alunos por se apresentar como uma plataforma inovadora, que apela à criatividade e imaginação dos alunos, que se adequa a uma aprendizagem baseada na resolução de problemas concretos cujos desafios criados promovem o raciocínio e o pensamento crítico de uma forma ativa, ou seja, promove contextos muito práticos de experimentação e resolução de desafios. O potencial pedagógico prometido pela Robótica Educativa é bastante promissor, se atendermos aos benefícios que lhe são apontados pela literatura existente. Não se pretendeu aqui demonstrar que a Robótica Educativa constitui uma moda para os problemas da Educação, mas é inegável que se trata de um recurso valioso que a escola dispõe e que já está demonstrado que tem fundamentos pedagógicos sérios e que contribui para o adquirir de competências em áreas nucleares.

Nesta investigação foi possível observar a interdisciplinaridade, o trabalho colaborativo e a aplicação do conhecimento em situações do mundo atual. Estas competências transversais fazem com que os alunos adquiram literacias múltiplas, necessárias aos desafios colocados à educação no quadro da sociedade atual, de elevada importância para o futuro do país.

É um trabalho resultante de diferentes atividades relacionadas com a eletricidade, a mecânica e a eletrónica em associação direta com a robótica, mas também com outros conteúdos do curriculum e com competências transversais, como a sustentabilidade ambiental, a educação para a cidadania, a promoção da igualdade de género e a inclusão.

6.1. Considerações finais

No presente momento, podemos observar que relativamente ao pensamento computacional, existem evidências teóricas e empíricas suficientes para afirmar a sua utilidade e validade, ou seja, estamos perante uma construção psicopedagógica que atingiu o ponto de maturidade.

Este estudo apresentou uma proposta pedagógica para a promoção do Pensamento Computacional no 1.º CEB através de atividades com a utilização da Robótica Educativa. Para tal, foi considerado um programa de intervenção desenhado com diversas atividades recorrendo a artefactos robóticos, que abordaram determinados conceitos básicos das ciências da computação. Apesar das limitações impostas pela sobreposição de outras atividades que conseqüentemente reduziram o tempo necessário para a concretização da proposta pedagógica, os resultados apontam no sentido de que os alunos aprenderam diversos conceitos das ciências da computação.

De acordo com os resultados obtidos nesta investigação, foi possível validar a presente metodologia para a utilização em sala de aula em turmas do 3.º e 4.º anos do 1.º CEB, uma vez que podemos considerar que os objetivos inicialmente definidos foram alcançados.

Entre os conjuntos robóticos, o Micro:Bit demonstrou ser o mais versátil e flexível, com diversas potencialidades e favorecendo a criatividade. Desempenhou um papel importante nesta investigação, possibilitando atividades STEAM, incentivando os alunos a projetar e construir como parte do processo de codificação. Especificamente, o Micro:Bit e a Plataforma MakeCode permitem uma progressão sustentada, fácil para qualquer pessoa que começa. Além disso, pode fornecer recursos avançados de desenvolvimento a um nível alto.

Acreditamos que as contribuições deste trabalho são diversas e poderão ser rentabilizadas quer a um nível mais global, quer em questões mais específicas. Assim, globalmente, destaca-se o trabalho de campo realizado e a metodologia encontrada, que juntos contribuíram para a confirmação de que a Robótica Educativa apresenta um contributo para o desenvolvimento do pensamento computacional no 1.º ciclo.

De facto, a contribuição deste trabalho estende-se quer no desenvolvimento de uma proposta de programa de intervenção, quer na descrição e caracterização dos materiais pedagógicos que suportam as atividades e na identificação dos níveis de desenvolvimento do pensamento computacional, através de um instrumento de avaliação.

Com base no exposto, espera-se que o presente trabalho possa ser útil para aqueles que desejam abordar o pensamento computacional e suas vicissitudes, bem como um ponto de partida para trabalhos futuros.

6.2. Limitações ao estudo

Em relação às limitações do estudo realizado, considera-se que provavelmente teria sido benéfico ao investigador ser professor de algumas turmas em estudo. Nesse caso, o investigador viveria a situação por dentro, sendo-lhe assim possível conhecê-la de forma mais profunda e registrar aspetos obtidos através da observação direta, enriquecendo a investigação. Por outro lado, o investigador está sempre dependente dos professores cooperantes relativamente a aplicação das atividades propostas. Embora não sendo o caso, poderá ter influência em investigações desta natureza.

Um outro aspeto que deve ser equacionado em futuras investigações é o número de sessões destinadas à intervenção pedagógica e a quantidade de artefactos robóticos a utilizar. Será prudente avaliar quais os temas e que tipo de materiais pedagógicos. Como resultado poderá ser necessário uma maior consistência pedagógica e, portanto, mais tempo para explorar/preparar.

Assim, com um maior número de aulas/horas, o uso de robôs poderia favorecer o trabalho de equipa, o que originaria um desenvolvimento mais autónomo e colaborativo de todos os alunos na consolidação das aprendizagens e outras competências do século XXI, particularmente no âmbito das ciências da computação.

Por fim, pelo exposto, e considerando os resultados bastante encorajadores associados aos conceitos relacionados com sequências, condições e ciclos simples, considera-se a pertinência de aprofundar este estudo, utilizando outro tipo de instrumento mais direcionado para as idades dos alunos desta investigação, por exemplo o trabalho de Zapata-Caceres et al. (2020).

Embora não fosse um objetivo desta investigação, num trabalho futuro, faz sentido introduzir o tema igualdade de género. Percecionar se existem diferenças significativas entre as respostas obtidas pelos rapazes e pelas raparigas.

No cômputo geral, existe um sentimento de dever cumprido sobre o trabalho desenvolvido, que apesar da pretensão do investigador, está longe de poder considerar-se um trabalho finalizado. Certamente seria utópico pensar de maneira diferente. Fica a percepção de que muitas questões ficaram por abordar,

mantendo o alerta e o desejo de prosseguir, um dia, com novas investigações sobre o pensamento computacional no ensino obrigatório.

6.3. Perspetivas futuras

Pretende-se dar continuidade ao trabalho realizado neste estudo, aplicá-lo a um público-alvo mais vasto, de forma a comprovar os resultados obtidos e as perceções resultantes deste estudo. Os resultados poderão melhorar assim como a fiabilidade do estudo.

Apesar de este estudo evidenciar ser promissor, espera-se ainda, como trabalho futuro, melhorar os recursos didáticos produzidos, incorporar sugestões dos professores titulares e responsáveis pela leção da disciplina de OCPR, alargar o número de horas da disciplina e propor uma estratégia que envolva o desenvolvimento das habilidades do pensamento computacional por meio de um conjunto de atividades progressivas.

Pretende-se propor, à direção e ao conselho pedagógico deste agrupamento, o alargamento do projeto, desde o ensino pré-escolar com atividades físicas *unplugged*, progredindo no 1.º ciclo (em todos os anos), acrescentando atividades com o uso de plataformas de codificação visual baseada em blocos e dispositivos robóticos (placa Micro:Bit) e, finalmente no 2.º ciclo com a programação baseada em texto.

Esta proposta estará ancorada numa abordagem holística, aproveitando os recursos Low Floor, High Ceiling (um conceito introduzido por Papert, nos anos 70 que pode ser consultado no endereço <https://mres.medium.com/designing-for-wide-walls-323bdb4e7277>) da placa Micro:Bit e da plataforma MakeCode. Será importante desenvolver e mobilizar o pensamento computacional, de acordo com as metas do programa do XXIII Governo e recomendações do CNE.

No entanto, se assumirmos que o pensamento computacional se baseia em conceitos, práticas e perspetivas típicas das ciências da computação, então o pensamento computacional necessariamente terá de evoluir paralelamente com essas ciências. Esse equilíbrio torna o construto intrinsecamente dinâmico e em constante progresso. Nesse sentido, será importante analisar e acompanhar futuramente a evolução do pensamento computacional, perante os recentes desenvolvimentos no ramo das ciências da computação como o da Inteligência Artificial no âmbito escolar.

“O caminho faz-se caminhando...”

7. REFERÊNCIAS BIBLIOGRÁFICAS

- Alsina, A., & Acosta, Y. (2022). Conectando la educación matemática infantil y el pensamiento computacional: aprendizaje de patrones de repetición con el robot educativo programable Cubetto®. *Innovaciones Educativas*, 24(37), 133–148. <https://doi.org/10.22458/IE.V24I37.4022>
- Anwar, S., Bascou, N. A., Menekse, M., & Kardgar, A. (2019). A systematic review of studies on educational robotics. In *Journal of Pre-College Engineering Education Research* (Vol. 9, Issue 2, pp. 19–42). Purdue University Press. <https://doi.org/10.7771/2157-9288.1223>
- Araujo, A. L. S. O., Serey Guerrero, D. D., Andrade, W. L., & Melo, M. R. A. (2019). How many abilities can we measure in computational thinking? A study on bebras challenge. *SIGCSE 2019 - Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 545–551. <https://doi.org/10.1145/3287324.3287405>
- Bakala, E., Gerosa, A., Hourcade, J. P., & Tejera, G. (2021). Preschool children, robots, and computational thinking: A systematic review. *International Journal of Child-Computer Interaction*, 29, 100337. <https://doi.org/10.1016/J.IJCCI.2021.100337>
- Baker, C. K., & Galanti, T. M. (2017). Integrating STEM in elementary classrooms using model-eliciting activities: responsive professional development for mathematics coaches and teachers. *International Journal of STEM Education*, 4(1). <https://doi.org/10.1186/s40594-017-0066-3>
- Bandura, A. (1993). Perceived Self-Efficacy in Cognitive Development and Functioning. *Educational Psychologist*, 28(2), 117–148. https://www.academia.edu/910085/Perceived_self_efficacy_in_cognitive_development_and_functioning
- Bandura, A. (1997). *Self-efficacy: the exercise of control* (1 ed.). New York: W. H. Freeman and Company.
- Barata, C., & Matos, J. F. (2019). Uso de objetos tangíveis programáveis na aprendizagem da programação. *Revista Intersaberes*, 14(31). <https://doi.org/10.22169/revint.v14i31.1505>
- Bardin, L. (2016). *Análise de conteúdo*. Edições 70. <https://madmunifacs.files.wordpress.com/2016/08/anc3a1lise-de-contec3bado-laurence-bardin.pdf>
- Barr, D., Harrison, J., & Conery, L. (2011). Computational Thinking: A Digital Age Skill for Everyone. *Learning & Leading with Technology*, 38(6), 20–23. <https://files.eric.ed.gov/fulltext/EJ918910.pdf>
- BBC - Make It Digital - About Make It Digital. (2022). <https://www.bbc.co.uk/programmes/articles/1gkwk58DPmRzt2TzDp3pr9x/about-make-it-digital>
- Berry, M. (2018, November 18). *Visual Programming – An open mind*. <http://milesberry.net/2018/11/visual-programming/>
- Bers, M. U., Strawhacker, A., & Sullivan, A. (2022). The state of the field of computational thinking in early childhood education. *OCDE Education Working Papers N. 274*. <https://doi.org/10.1787/3354387a-en>
- Brackmann, C. P. (2017). *Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica* [Master's thesis, Universidade Federal do Rio Grande do Sul Centro]. <https://doi.org/10.13140/RG.2.2.32976.61444>
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Annual American Educational Research Association Meeting, Vancouver, BC, Canada, 2012*, 1–25. http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf
- Bruner, J. S. (1961). The act of discovery. *Harvard Educational Review*, 31, 21–32. <https://hal.archives-ouvertes.fr/hal-00692072>
- Chalmers, C. (2018). Robotics and computational thinking in primary school. *International Journal of Child-Computer Interaction*, 17, 93–100. <https://doi.org/10.1016/J.IJCCI.2018.06.005>
- Chiazesse, G., Arrigo, M., Chifari, A., Lonati, V., & Tosto, C. (2018, October 24-26). *Exploring the Effect of a Robotics Laboratory on Computational Thinking Skills in Primary School Children Using the Bebras Tasks*. Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality, Salamanca, Spain. <https://doi.org/10.1145/3284179.3284186>
- Chou, P.-N. (2018, November 12-16). *Little Engineers: Young Children's Learning Patterns in an Educational Robotics Project*. 2018 World Engineering Education Forum - Global Engineering Deans Council (WEEF-GEDC), Albuquerque, NM, USA. <https://doi.org/10.1109/WEEF-GEDC.2018.8629609>
- Çoban, E., & Korkmaz, Ö. (2021). An alternative approach for measuring computational thinking: Performance-based platform. *Thinking Skills and Creativity*, 42, 100929. <https://doi.org/10.1016/j.tsc.2021.100929>
- Coutinho, C. M. P. (2022). *Metodologia de Investigação em Ciências Sociais e Humanas: Teoria e Prática* (2nd ed.). Almedina. <https://www.almedina.net/metodologia-de-investiga-o-em-ci-ncias-sociais-e-humanas-teoria-e-pr-tica-1563799552.html>
- Creative Computing Lab. (2021). *Creative Computing Curriculum | Overview*. <https://creativecomputing.gse.harvard.edu/guide/index.html>
- Creswell, J. W. (2014). *Research design: qualitative, quantitative and mixed methods approaches* (p. 275). Sage.
- Cristina, A., & Rocha, A. (2020). *A plataforma Arduino no apoio ao desenvolvimento dos projetos interdisciplinares dos cursos profissionais* [Master thesis, Universidade do Minho]. Repositório Institucional da UM. <https://repositorium.sdum.uminho.pt/handle/1822/68582>
- CSTA. (2016). *K-12 Computer Science Framework*. <http://www.k12cs.org>.
- CSTA. (2017). Progression of Computer Science Teachers Association (CSTA) K-12 Computer Science Standards, Revised 2017. *Csta*, 3017. <http://www.csteachers.org/standards>.

- Demetriadis, S., Atmatzidou, S., & Sapounides, T. (2012, April 20). *The AUTH Framework for Research in Educational Robotics : Collaboration Scripts , Metacognitive Skills , Tangible Interfaces and the CPPC + Model T*. 3rd International Workshop Teaching Robotics, Teaching with Robotics Integrating Robotics in School Curriculum Riva del Garda, Trento, Italy. https://www.terecop.eu/TRTWR2012/zz_trtwr2012_submission_18.pdf
- Denzin, N. K., & Lincoln, Y. S. (2017). *The Sage handbook of qualitative research* (5th ed.). SAGE Publications, Inc.
- Diago, P., Vera, D., & Somoza, J. (2018). Elementos de resolución de problemas en primeras edades escolares con Bee-bot. *Edma 0-6: Educación Matemática En La Infancia*, 7(1), 12–41. <https://doi.org/10.24197/edmain.1.2018.12-41>
- Donaldson, J. (2014, January 23). *The maker movement and the rebirth of constructionism*. Digital Pedagogy. <https://hybridpedagogy.org/constructionism-reborn/>
- Fantinati, R. E., & Rosa, S. D. S. (2021). Pensamento Computacional: Habilidades, Estratégias e Desafios na Educação Básica. *Informática Na Educação: Teoria & Prática*, 24(1 Jan/Abr). <https://doi.org/10.22456/1982-1654.110751>
- Franklin, D., Coenraad, M., Palmer, J., EATINGER, D., Zipp, A., Anaya, M., White, M., Pham, H., Gökdemir, O., & Weintrop, D. (2020, August 10–12). *An Analysis of Use-Modify-Create Pedagogical Approach's Success in Balancing Structure and Student Agency*. Proceedings of the 2020 ACM Conference on International Computing Education Research (ICER'20), Virtual Event, New Zealand. <https://doi.org/10.1145/3372782.3406256>
- Freixo, M. (2011). *Metodologia científica : fundamentos, métodos e técnicas*. Instituto Piaget.
- Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015, March 18-20). *Computing education in K-12 schools: A review of the literature*. 2015 IEEE Global Engineering Education Conference (EDUCON), Tallinn University of Technology, Tallinn, Estonia. <https://doi.org/10.1109/EDUCON.2015.7096023>
- Gil, A. C. (1999). *Métodos e técnicas de pesquisa social* (5th ed.). Atlas.
- Gonçalves, A., & Freire, C. (2012, November 30-December 2). *O primeiro ano do projecto de robótica educativa*. TICEDUCA2012 - Atas Do II Congresso Internacional TIC e Educação, Faculdade de Psicologia e de Ciências da Educação, Lisboa - Portugal. https://cidfff.web.ua.pt/producao/dayse_souza/atas_ticEduca.pdf
- González, M. R. (2016). *Codigofabetización y pensamiento computacional en Educación Primaria y Secundaria: validación de un instrumento y evaluación de programas* [Master 's thesis, Universidad Nacional de Educación a Distancia]. <http://e-spacio.uned.es/fez/view/tesisuned:Educacion-Mroman>
- Guedes, A., Souza, G., & Junior, A. (2021). (PDF) O uso da robótica no Desenvolvimento do Pensamento Computacional: uma Revisão Sistemática. *Editora E-Publicar*. https://www.researchgate.net/publication/355781974_O_uso_da_Robotica_no_Desenvolvimento_do_Pensamento_Computacional_uma_Revisao_Sistemtica
- Hodges, S., Sentance, S., Finney, J., & Ball, T. (2020). Physical Computing: A Key Element of Modern Computer Science Education. *Computer*, 53(4), 20–30. <https://doi.org/10.1109/MC.2019.2935058>
- Huang, W. Y., Hu, C. F., & Wu, C. C. (2018). The Use of Different Kinds of Robots to Spark Student Interest in Learning Computational Thinking. *Proceedings - 2018 6th International Conference on Learning and Teaching in Computing and Engineering, LaTICE 2018*, 11–16. <https://doi.org/10.1109/LaTICE.2018.00-13>
- ISTE, & CSTA. (2011). *Computational Thinking leadership toolkit*. https://cdn.iste.org/www-root/2020-10/ISTE_CT_Leadership_Toolkit_booklet.pdf
- Kiang, M., & Kiang, D. (2019). *Making with MakeCode & Micro:bit*. 1–229. <https://makecode.com/blog/microbit/csintro-educator>
- Kim, D., & Bolger, M. (2017). Analysis of Korean Elementary Pre-Service Teachers' Changing Attitudes About Integrated STEAM Pedagogy Through Developing Lesson Plans. *International Journal of Science and Mathematics Education*, 15(4), 587–605. <https://doi.org/10.1007/s10763-015-9709-3>
- Körber, N., Bailey, L., Greifenstein, L., Fraser, G., Sabitzer, B., & Rottenhofer, M. (2021). An Experience of Introducing Primary School Children to Programming using Ozobots (Practical Report). *The 16th Workshop in Primary and Secondary Computing Education*, 1–6. <https://doi.org/10.1145/3481312.3481347>
- Krueger, R. A., & Casey, M. A. (2015). *Focus Groups: A Practical Guide for Applied Research* (5th ed.). SAGE Publications, Inc.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1), 32–37. <https://doi.org/10.1145/1929887.1929902>
- Lego, E. (2016). *LEGO Education WeDo 2.0: Projetos Curriculares*. The LEGO Group. <https://education.lego.com/v3/assets/blt293eea581807678a/bltbdad1a20672a97a2/5ebe9629f8b8c35280daee5a/br-wedo-introduction.pdf>
- Lego, E. (2017). *WeDo 2.0 Support | Everything You Need | LEGO® Education*. <https://education.lego.com/en-gb/product-resources/wedo-2/teacher-resources/teacher-guides#english-us>
- Lehtimäki, T., Monahan, R., Mooney, A., Casey, K., & Naughton, T. J. (2022, July 8–13). *Bebras-inspired Computational Thinking Primary School Resources Co-created by Computer Science Academics and Teachers*. Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education (ITICSE 2022), Dublin, Ireland. <https://doi.org/10.1145/3502718.3524804>
- Lopes, M. de M., Branco, V. T., & Soares, J. B. (2013). Utilização dos testes estatísticos de Kolmogorov-Smirnov e Shapiro-Wilk para verificação da normalidade para materiais de pavimentação. *Transportes*, 21(1), 59. <https://doi.org/10.4237/transportes.v21i1.566>
- Marques, A. S. (2021). *Iniciação à Programação no 1.º Ciclo do Ensino Básico: Análise da implementação da medida a alunos do 3.º ano* [Master 's thesis,

- Universidade Beira Interior]. uBibliorum - Repositório digital da UBI. <http://hdl.handle.net/10400.6/11156>
- Mattar, J., & Ramos, D. K. (2021). *Metodologia da Pesquisa em educação: abordagens qualitativas, quantitativas e mistas* (Vol. 1). Grupo Almedina. https://play.google.com/store/books/details/Metodologia_da_pesquisa_em_educacao_Abordagens_Qua?id=tF49EAAQBAJ&gl=US
- MEC. (2021). *Despacho n.º 8209/2021 / DRE*. <https://dre.pt/dre/detalhe/despacho/8209-2021-169831748>
- Meirinhos, M., & Osório, A. J. (2010). O estudo de caso como estratégia de investigação em educação. *EDUSER: Revista de Educação*, 2(2), 49–65. <https://doi.org/https://doi.org/10.34620/eduser.v2i2.24>
- Mester, G. (2016). Massive Open Online Courses in Education of Robotics. *Interdisciplinary Description of Complex Systems*, 14(2), 182–187. <https://doi.org/10.7906/INDECS.14.2.7>
- Miles, M. B., & Huberman, A. M. (1994). Qualitative Data analysis A Source Book of New Methods. In *Beverly Hills: Sage*.
- Montessori, M. (2007). *From childhood to adolescence*. Montessori-Pierson.
- Morgan, D. L., & Kruger, R. A. (1998). *The Focus Group Guide Book*. Sage Publications, Inc. <https://books.google.co.za/books?id=nLdCQAAQBAJ&dq=morgan+kruger+1998+focus+group&lr=>
- Neto, J., & Bertagnolli, S. (2021). Robótica educacional e formação de Professores: Uma revisão sistemática da literatura. *RENOTE*, 19(1), 423–432. <https://doi.org/10.22456/1679-1916.118532>
- Obama, B. (2013). Don't Just Play on Your Phone, Program It. *The White House Blog*. <https://obamawhitehouse.archives.gov/blog/2013/12/09/don-t-just-play-your-phone-program-it>
- Papert, S. (1980). Mindstorms: Children, Computers, and Powerful Ideas. In *Basic Books, Inc*. <https://doi.org/https://doi.org/10.2307/816450>
- Papert, S. (1986). Constructionism: a new opportunity for elementary science education. In *The National Science Foundation*. Massachusetts Institute of Technology Media Laboratory Epistemology and Learning Group. <http://dailypapert.com/wp-content/uploads/2021/02/Constructionism-NSF-Proposal.pdf>
- Papert, S. (1993). *The children's machine: rethinking school in the age of the computer*. Basic-Books.
- Pedro, A., Matos, F. J., Piedade, J., & Dorotea, N. (2017). Probótica Programação e robótica no Ensino Básico - Linhas Orientadoras. *Instituto de Educação Da Universidade de Lisboa*, 38.
- Piaget, J. (1986). *O nascimento da inteligência na criança*. Dom Quixote. <http://hdl.handle.net/10400.2/7599>
- Pinto, M., Fernandes, M., & Osório, A. J. (2021, March 8-9). *Curriculum integration proposal for kibo robot in preschool: kids media lab project*. INTED2021 15th International Technology, Education and Development Conference, Online Conference. <https://doi.org/10.21125/inted.2021.1591>
- Rachmatullah, A., Mayhorn, C. B., & Wiebe, E. N. (2021). The effects of prior experience and gender on middle school students' computer science learning and monitoring accuracy in the Use-Modify-Create progression. *Learning and Individual Differences*, 86, 101983. <https://doi.org/10.1016/J.LINDIF.2021.101983>
- Ramos, J., & Espadeiro, R. (2014a). Introdução do pensamento computacional na formação inicial de professores - Questões de avaliação e investigação. *CIAIQ2014*, 3. <https://proceedings.ciaiq.org/index.php/CIAIQ/article/view/485>
- Ramos, J., & Espadeiro, R. (2014b). Os futuros professores e os professores do futuro. Os desafios da introdução ao pensamento computacional na escola, no currículo e na aprendizagem. *Educação, Formação e Tecnologias*, 7(2), 4–25. <http://eft.educom.pt/index.php/eft/article/view/462>
- Ramos, J. L., & Espadeiro, R. G. (2015 Maio 14-15). *Pensamento computacional na escola e práticas de avaliação das aprendizagens. Uma revisão sistemática da literatura*. Challenges 2015 : Meio século de TIC na Educação, Universidade do Minho, Braga, Portugal. <http://hdl.handle.net/10174/18147>
- Razali, N. M., & Wah, Y. B. (2011). Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. *Journal of Statistical Modeling and Analytics*, 2, 21–33. <https://www.researchgate.net/publication/267205556>
- Reis, E. (1996). *Estatística aplicada*. Lisboa: Edições Sílabo.
- Relkin, E., & Bers, M. (2021, April 21-23). *TechCheck-K: A Measure of Computational Thinking for Kindergarten Children*. 2021 IEEE Global Engineering Education Conference (EDUCON), 1696–1702, Vienna, Austria. <https://doi.org/10.1109/EDUCON46332.2021.9453926>
- Resnick, M. (2007, June 13-15). *All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten*. Proceedings of the 6th ACM SIGCHI Conference on Creativity & Cognition - C&C '07. Washington, DC, USA. <https://doi.org/10.1145/1254960.1254961>
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60–67. <https://doi.org/10.1145/1592761.1592779>
- Rich, K. M., Binkowski, T. A., Strickland, C., & Franklin, D. (2018, August 13–15). *Decomposition: A K-8 Computational Thinking Learning Trajectory*. Proceedings of the 2018 ACM Conference on International Computing Education Research (ICER), Espoo, Finland. <https://doi.org/10.1145/3230977.3230979>
- Rodrigues, M. R., & Felício, P. (2019, November 21-23). *The use of ground robots in primary education: students' perspectives*. 2019 International Symposium on Computers in Education (SIE), Tomar, Portugal. <https://doi.org/10.1109/SIE48397.2019.8970141>

- Rogers, M. P., & Siever, B. (2018). A Macro View of the Micro: Bit in Higher Education. *J. Comput. Sci. Coll.*, 33(5), 124–132. <https://doi.org/10.5555/3204979>
- Román-González, M., Moreno-León, J., & Robles, G. (2019). Combining Assessment Tools for a Comprehensive Evaluation of Computational Thinking Interventions. In *Computational Thinking Education* (pp. 79–98). Springer Singapore. https://doi.org/10.1007/978-981-13-6528-7_6
- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*, 72, 678–691. <https://doi.org/10.1016/J.CHB.2016.08.047>
- Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Can computational talent be detected? Predictive validity of the Computational Thinking Test. *International Journal of Child-Computer Interaction*, 18, 47–58. <https://doi.org/10.1016/J.IJCCI.2018.06.004>
- Safarati, N., & Rahma, R. (2020). The Effectiveness of Online Learning Using Quizizz Education Game Media During the Covid-19 Pandemic in Applied Physics Courses. *Indonesian Review of Physics*, 3(2), 52. <https://doi.org/10.12928/IRIP.V3I2.3049>
- Sentance, S., Waite, J., Yeomans, L., & MacLeod, E. (2017, November 8-10). *Teaching with physical computing devices: the BBC micro:bit initiative*. Proceedings of the 12th Workshop on Primary and Secondary Computing Education (WiPSCE), Nijmegen, Netherlands. <https://doi.org/10.1145/3137065.3137083>
- Sousa, K. F., & Araújo, N. M. (2021). Quizizz nas aulas de inglês como L2. *Ilha Do Desterro A Journal of English Language, Literatures in English and Cultural Studies*, 74(3), 161–182. <https://doi.org/10.5007/2175-8026.2021.e80734>
- Stamatios, P. (2022). Can Preschoolers Learn Computational Thinking and Coding Skills with ScratchJr? A Systematic Literature Review. *International Journal of Educational Reform*, 0, 1–34. <https://doi.org/10.1177/10567879221076077>
- Tardif, M. (2010). *Saberes docentes e formação profissional*. Vozes.
- Torman, V. B. L., Coster, R., & Riboldi, J. (2012). Normalidade de variáveis: métodos de verificação e comparação de alguns testes não-paramétricos por simulação. *Clinical and Biomedical Research*, 32(2). <https://seer.ufrgs.br/index.php/hcpa/article/view/29874>
- Trundle, K. C., & Smith, M. M. (2017). A Hearts-on, Hands-on, Minds-on Model for Preschool Science Learning. *YC Young Children*, 72(1), 80–86. https://www.academia.edu/31988916/A_Hearts_on_Hands_on_Minds_on_Model_for_Preschool_Science_Learning
- Üşengül, L., & Bahçeci, F. (2020). The Effect of Lego Wedo 2.0 Education on Academic Achievement and Attitudes and Computational Thinking Skills of Learners toward Science. *World Journal of Education*, 10(4), 83. <https://doi.org/10.5430/wje.v10n4p83>
- Valente, J. (2016). do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista E-Curriculum*, 14(3), 864–897. <https://www.redalyc.org/pdf/766/76647706006.pdf>
- Vilelas, J. (2020). *Investigação o processo de construção do conhecimento* (3rd ed.). Edições Sílabo.
- Wilkinson, S. (2004). Focus groups: A feminist method. In S. N. Hesse-Biber & M. Yaiser (Eds.), *Feminist perspectives on social research*. (pp. 271–295). Oxford: University Press.
- Wing, J. (2014, February 7). *Computational Thinking*. Columbia Journalism School. https://archive.org/details/podcast_columbia-journalism-school_computational-thinking-with-je_1000250903549
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2017). Computational thinking 's influence on research and education for all. *Italian Journal of Educational Technology*, 25(2), 7–14. <https://doi.org/https://doi.org/10.17471/2499-4324/922>
- Wing, J. M., & Stanzione, D. (2016). Progress in computational thinking, and expanding the HPC community. *Communications of the ACM*, 59(7), 10–11. <https://doi.org/10.1145/2933410>
- Yin, R. K. (2015). *Estudo de caso. Planejamento e métodos*. (5). Porto Alegre: Bookman.
- Žáček, M., & Smolka, P. (2019, November 5-7). *Development of computational thinking: Student motivation using Ozobot*. Proceedings of the 2019 3rd International Conference on Education and E-Learning (ICEEL 2019), Barcelona, Spain. <https://doi.org/10.1145/3371647.3371654>
- Zapata-Caceres, M., Martin-Barroso, E., & Roman-Gonzalez, M. (2020, April 27-30). *Computational Thinking Test for Beginners: Design and Content Validation*. 2020 IEEE Global Engineering Education Conference (EDUCON), Porto, Portugal. <https://doi.org/10.1109/EDUCON45650.2020.9125368>
- Zhao, W., & Shute, V. J. (2019). Can playing a video game foster computational thinking skills? *Computers & Education*, 141, 103633. <https://doi.org/10.1016/j.compedu.2019.103633>

8. ANEXOS

Anexo A – Autorização de Marcos Román González para a utilização e adaptação do seu teste de pensamento computacional

Anexo B – Autorização do MIME

Anexo C1 – Autorização dos Encarregados de Educação

Anexo C2 – Autorização da direção do agrupamento Fernando Távora para a realização do Teste Piloto

Anexo D – Poster apresentação nos REDs do Centro de Formação da Escola Secundária

Anexo E1 – Programa de intervenção – sessão 1 – Apresentação do conjunto Lego Wedo 2.0

Anexo E2 – Programa de intervenção – sessão 2 – Milo, o Robô Explorador da Ciência

Anexo E3 – Programa de intervenção – sessão 3 – Apresentação da placa Micro:Bit

Anexo E4 – Programa de intervenção – sessão 4 – Exploração de variáveis com Micro:Bit

Anexo E5 – Programa de intervenção – sessão 5 – Testar condições com a placa Micro:Bit

Anexo E6 – Programa de intervenção – sessão 6 – O que são Iterações?

Anexo E7 – Programa de intervenção – sessão 7 – Mini-Projeto “Hot Potato”

Anexo E8 – Programa de intervenção – sessão 8 – Mini-Projeto “Buzzer Game”

Anexo E9 – Programa de intervenção – sessão 9 – Apresentação do Robô Ozobot, programação por cores

Anexo E10 – Programa de intervenção – sessão 10 – Introdução ao Ozobot Blockly

Anexo F – Painel de sessão Recursos Educativos Digitais

Anexo G – Guião *focus group*

Anexo H - Teste de Pensamento Computacional aplicado em formato digital

Anexo I - Resultados obtidos

Anexo A - Autorização de Marcos Román González e Christian Brackman para a utilização e adaptação do seu teste de pensamento computacional

24/05/22, 22:06 Email - José Carlos Pereira da Silva - Outlook

RE: Solicitação para a aplicação da ferramenta de teste do Pensamento Computacional

MARCOS ROMAN GONZALEZ <mroman@edu.uned.es>
Mon 14/02/2022 11:54
To: José Carlos Pereira da Silva <pg42441@alunos.uminho.pt>

Buenas tardes José Carlos,

Con mucho gusto, si autorizamos y compartimos libremente el Test de Pensamiento Computacional con fines de investigación...

...en qué idioma le interesa: ¿español, inglés, portugués...?

Por favor, confírmeme el idioma que desea.

Saludos cordiales.



Facultad
de Educación

Marcos Román González
Coordinador de Comunicación, Virtualización y Tecnologías
Facultad de Educación, Departamento MIDE I

C/ Juan del Rosal, 14, Despacho 2.18. 28040 Madrid
T. +34 91 398 90 37
mroman@edu.uned.es



#SOMOS2030

24/05/22, 22:08

Email - José Carlos Pereira da Silva - Outlook

Re: Solicitação para a aplicação da ferramenta de teste do Pensamento Computacional

Christian Brackmann <brackmann@ifar.edu.br>

Fri 18/02/2022 19:47

To: José Carlos Pereira da Silva <pg42441@alunos.uminho.pt>

Prezado Sr. José Carlos Silva.

Peço desculpas pela demora da resposta. Prazer em receber o seu contato.

Recomendo a leitura da minha tese: <https://www.lume.ufrgs.br/handle/10183/172208>. Uma cópia do teste do Prof. Marcos em Português encontra-se no Apêndice A e as respostas no Anexo F. Caso tenha interesse em acessar a versão online, você pode entrar em contato por e-mail comigo para lhe enviar uma cópia pelo Google Drive Forms.

Sem mais para o momento, despeço-me reiterando votos de elevada estima e distinta consideração.

Qualquer dúvida, encontro-me à disposição.

Atenciosamente,

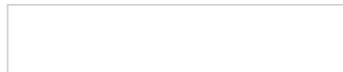
Prof. Christian Puhlmann Brackmann, Dr.

Pró-reitor de Extensão em Exercício - Portaria 648/2020

Instituto Federal de Educação, Ciência e Tecnologia Farroupilha (IFFAR)

brackmann@ifar.edu.br

+55 55 3218 9800 / 98100-1819



Anexo B – Autorização do MIME

Monotorização de Inquéritos em Meio Escolar: Inquérito nº 0838700001

mime.noreply@min-educ.pt <mime.noreply@min-educ.pt>

Tue 12/07/2022 17:07

To: José Carlos Pereira da Silva <pg42441@alunos.uminho.pt>; António José Meneses Osório <ajosorio@ie.uminho.pt>

Exmo(a)s. Sr(a)s.

O pedido de autorização do inquérito n.º 0838700001, com a designação *Projeto de Dissertação de Mestrado: "CONTRIBUTO PARA O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL NA ESCOLARIDADE INICIAL"*, registado em 25-05-2022, foi aprovado.

Avaliação do inquérito:

Exmo.(a) Senhor(a) José Carlos Pereira da Silva

Venho por este meio informar que o pedido de realização de inquérito em meio escolar é aprovado uma vez que, submetido a análise, cumpre os requisitos, devendo atender-se às observações aduzidas.

Com os melhores cumprimentos

José Vítor Pedroso

Diretor-Geral

DGE

Observações:

- a) A realização dos Inquéritos fica sujeita a autorização das Direções dos Agrupamentos de Escolas do ensino público a contactar para a realização do estudo. Merece especial atenção o modo, o momento e condições de aplicação dos instrumentos de recolha de dados em meio escolar, porque onerosos, devendo fazer-se em estreita articulação com as Direções dos Agrupamentos e com os encarregados de educação ou quem tutele os menores inquiridos.
- b) Deve considerar-se o disposto legal em matéria de garantia de anonimato dos sujeitos, não identificabilidade, confidencialidade, proteção e segurança dos dados pessoais a recolher e tratar no presente estudo. Considerados os documentos que foram anexados e para efeitos da proteção dos dados pessoais a recolher junto dos inquiridos, em cumprimento da legislação em vigor, resultam obrigações que o responsável se propõe cumprir. Destas deve dar conhecimento a todos os inquiridos e a quem intervenha na recolha e tratamento de dados. É obrigatória a recolha prévia das declarações de consentimento inequívoco, informado e esclarecido, junto dos inquiridos, titulares dos dados, no caso de menores, junto dos encarregados de educação ou quem tutele os menores inquiridos. As autorizações assinadas devem ficar em poder da Escola/Agrupamento ao qual pertencem. Não deve haver cruzamento ou associação de dados entre os que são recolhidos pelos instrumentos de inquirição e os constantes das declarações de consentimento informado. Recomenda-se que, dado o exposto, para efeitos de proteção de dados e cumprimento do disposto legal nesta matéria, o/a Encarregado/a de Proteção de Dados da entidade

responsável pelo estudo possa apoiar todo o processo. Sugere-se, ainda, que para efetiva garantia do anonimato, o nome do inquirido e a designação do estabelecimento de educação/ensino possa ser substituído por um código. E bem assim, a declaração de consentimento informado anexado possa ser revista (ie: Ao longo da investigação serão recolhidos registos fotográficos, registo em vídeo), ponderando-se acionar medidas de salvaguarda previstas na lei para segurança dos mesmos e devida proteção dos titulares.

Pode consultar na Internet toda a informação referente a este pedido no endereço <http://mime.dgeec.mec.pt>. Para tal terá de se autenticar fornecendo os dados de acesso da entidade.

Anexo C1 – Autorização dos Encarregados de Educação

Exmo(a). Sr(a).

Encarregado(a) de Educação

O meu nome é José Carlos Silva e sou docente no Agrupamento de Escolas, em Guimarães além de ser estudante do mestrado em Mestrado em Ciências da Educação Área de Especialização em Tecnologia Educativa lecionado pela Universidade do Minho. De momento estou a desenvolver uma Investigação cujo tema é: “**ROBÓTICA EDUCATIVA, UM CONTRIBUTO PARA O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL NA ESCOLARIDADE INICIAL**”, com o objetivo de procura caracterizar o processo de desenvolvimento de atividades de Robótica Educativa com crianças do 1.º ciclo do ensino básico, que promovam dimensões de Pensamento Computacional.

A referida investigação tem como orientador o professor Doutor António José Meneses Osório.

Para tal, gostaria de solicitar a sua autorização e colaboração para recolher dados relativos ao seu educando e seu processo de ensino aprendizagem mais concretamente no que envolva a disciplina de Oferta Complementar de Programação e Robótica (OCPR) deste agrupamento.

Saliento ainda que o estudo não envolve qualquer alteração no tratamento estabelecido ao longo das aulas e que a recolha de dados será feita mediante a análise dos registos bibliográficos dos alunos e dos resultados obtidos na realização de diferentes tarefas que constam nos planos de aula.

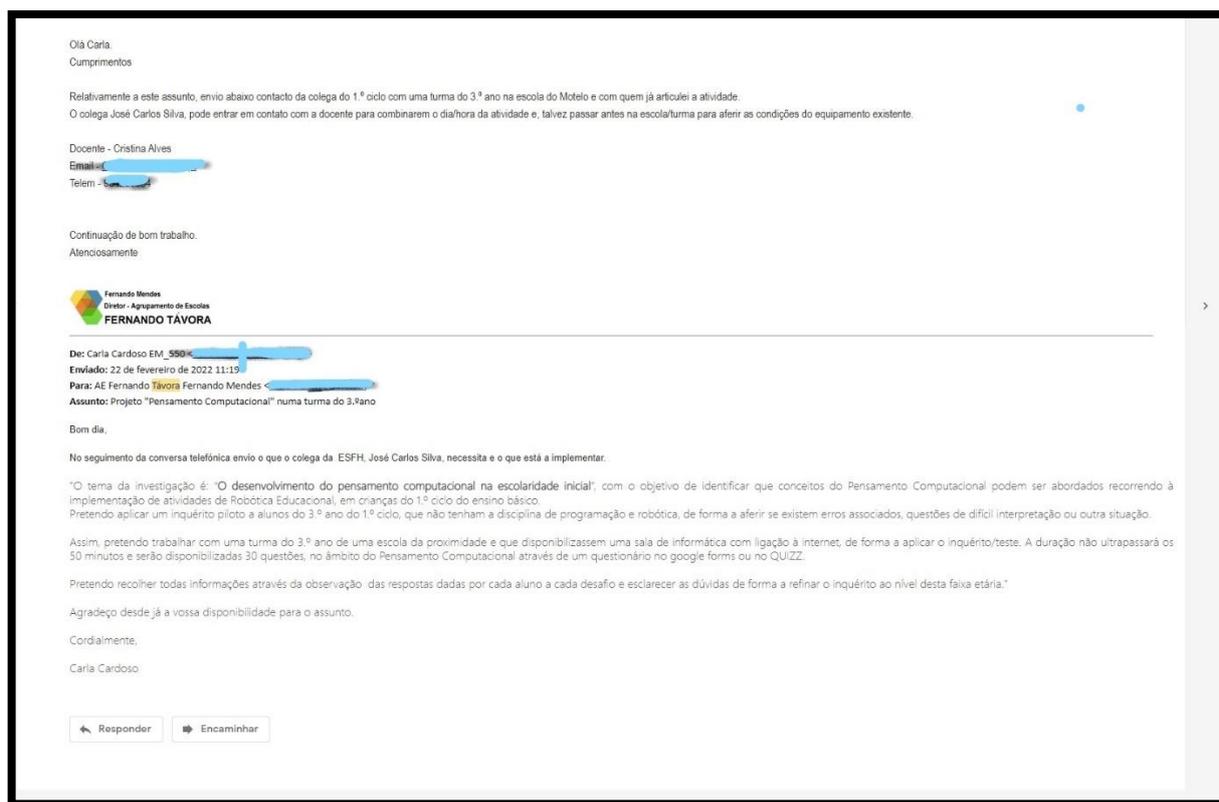
Certo que a sua autorização me irá ajudar a desenvolver este estudo, agradeço antecipadamente a sua colaboração e disponibilidade.

✂

PEDIDO DE AUTORIZAÇÃO AO ENCARREGADO DE EDUCAÇÃO

Eu _____, Encarregado(a) de Educação do aluno (a) _____, autorizo/não autorizo a recolha de dados no âmbito do estudo de investigação ‘ROBÓTICA EDUCATIVA CONTRIBUTO PARA O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL NA ESCOLARIDADE INICIAL’.

Anexo C2 – Autorização da direção do Agrupamento Fernando Távora para a realização do Teste Piloto



Anexo D – Poster apresentação nos REDs do Centro de Formação da Escola Secundária

Objetivos Gerais da Ação	Recursos Educativos Digitais (RED)	Organização
<p><i>A Capacitação Digital das Escolas pretende impulsionar o desenvolvimento de estratégias inovadoras e indutoras de mudança que terão impacto na melhoria da gestão escolar, na qualidade do ensino, nas aprendizagens e nos resultados escolares. (DGE)</i></p> <ul style="list-style-type: none"> • Instituir práticas colaborativas facilitadoras da aprendizagem; • Analisar recursos educativos digitais de suporte ao PADDE; • Desmistificar conceitos para transformar práticas; • Promover comunidades de prática, para estimular a reflexão e a colaboração em contexto educativo; • Refletir sobre processos organizativos, pedagógicos e tecnológicos que, com recurso a ferramentas e a ambientes digitais, promovam a qualidade do processo educativo. 	<p>Com vista a cumprir os desígnios da DGE, o CFFH propõe uma sessão formativa/informativa, com o objetivo geral de promover Recursos Educativos Digitais de qualidade, de acesso livre, validados científica e pedagogicamente, e de apoiar o seu desenvolvimento nas escolas da sua área geográfica.</p> <p>A utilização destes recursos contribuirá para:</p> <ul style="list-style-type: none"> • A inovação e a diferenciação pedagógica; • A adoção de estratégias promotoras do desenvolvimento crítico, criativo e autónomo dos alunos; • O incentivo a modos de avaliação que promovam a aprendizagem (avaliação pedagógica); • A indução de práticas de sala de aula que melhor atendam à prossecução do perfil dos alunos à saída da escolaridade obrigatória. 	<p>Conjunto de duas sessões de 1h30m cada, para constituir uma ação de curta duração de 3h, com requisitos para ser reconhecida, nos termos do Despacho n.º 5741/2015, de 29 de maio:</p> <p>05 de julho—18h – 19h30 Apresentação dos Recursos Educativos Digitais disponíveis na plataforma da DGE.</p> <p>19 de julho—18h – 19h30 Apresentação de práticas com utilização de RED:</p> <ul style="list-style-type: none"> • AE Pevidém — Professoras Maria da Conceição Bessa e Cristina Almeida (Projetos dinamizados no 1º CEB) • AE Fafe— Renato Silva / Aprendizagem invertida com grupo de especialistas: um exemplo (Projetos dinamizados no 2º CEB) • AE Francisco de Holanda— Professores José Carlos Silva Sandra Machado e Carla Barbosa (Projetos dinamizados no 1º CEB) • AEPAS—Professor Luís Silva (Projetos dinamizados no 1º CEB) <p>Destinatários Professores das escolas associadas ao CFFH.</p>

Anexo E1 – Programa de intervenção – sessão 1 – Apresentação do conjunto LEGO Wedo 2.0

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	<p>SESSÃO N.º 1 LEGO WEDO 2.0</p> <p>OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)</p> <p>02/2022 - 50 min.</p>
	<p>Ano Letivo 2021/2022</p>

Objetivos: Dar a conhecer os principais blocos constituintes do conjunto Lego Wedo 2.0. Exercício global de demonstração passo a passo, de acordo com as fases: Explora, Criar e Compartilhar.

1. O trabalho a realizar, será em grupo de dois alunos. Cada grupo vai buscar um conjunto Lego Wedo 2.0 disponibilizado pela professora.
2. Certifica-te que também tens um tablet ou um computador.
3. Abre o software/programa da Lego Wedo 2.0. Está atenta(o) à explicação da professora.
4. Agora visualiza os seguintes vídeo com a tua professora sobre alguns procedimentos importantes a ter em conta com o kit da Lego.



Inserir as pilhas/baterias	https://education.lego.com/en-gb/start/wedo-2#Insert-Batteries
Identificar cada conjunto de peças	https://education.lego.com/en-gb/start/wedo-2#Apply-Stickers
Ordenar/separar as peças do conjunto	https://education.lego.com/en-gb/start/wedo-2#Sort-the-Bricks
Numerar as peças mais importantes	https://education.lego.com/en-gb/start/wedo-2#Number-the-Items

Ligação do SmartHub ao tablet ou computador

5. Nesta fase vais aprender a fazer a ligação do Hub ao tablet/computador para poderes programar.
6. Identifica as seguintes partes que estão no kit do Wedo 2.0.

SmartHub	Sensor de inclinação	Sensor de movimento	motor
			



7. Verifica se tens pilhas, dentro do Hub.

8. Verifica se o *bluetooth* está ligado no tablet ou portátil. Tenta emparelhar o *Smart Hub* com o tablet ou portátil. Carrega no botão verde do *HUB* para o ligares (o led fica iluminado a branco e a piscar).
9. Abre a aplicação do *Wedo 2.0* e cria um projeto em branco (sinal +) em meus projetos.
10. Nos dispositivos *bluetooth*, tenta emparelhar o *SmartHub* ao tablet ou computador, devem aparecer vários, no entanto apenas o número que está atribuído ao teu grupo deve ser selecionado. Caso não estejas a conseguir chama a tua professora.
11. Agora que já tens *SmartHub* a aparecer no software ou aplicação será boa prática, alterar o nome para o número que identifica o teu *SmartHub*. Deve estar identificado com um número.



Projeto inicial – Caracol incandescente

12. Neste projeto a atividade proposta, passa por:
 - a. Constrúes um modelo Lego (*Caracol*);
 - b. Ligares o modelo ao dispositivo (*SmartHub*);
 - c. Programar o caracol para mostrar várias cores;
 - d. Explora o programa proposto e propõe outras situações.
13. Na aplicação do *Wedo 2.0*, procura o projeto introdutório *Caracol incandescente*.



Projeto Introdutório

Caracol incandescente



14. Segue todos os passos apresentados. Caso tenhas dúvidas pergunta aos teus colegas ou à professora.

Bom trabalho 😊

Anexo E2 – Programa de intervenção – sessão 2 – Milo, o Robô Explorador da Ciência

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	SESSÃO N.º 1 LEGO WEDO 2.0
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR) 02/2022 - 50 min. Ano Letivo 2021/2022

Objetivos: Construir um robô capaz de explorar lugares remotos.

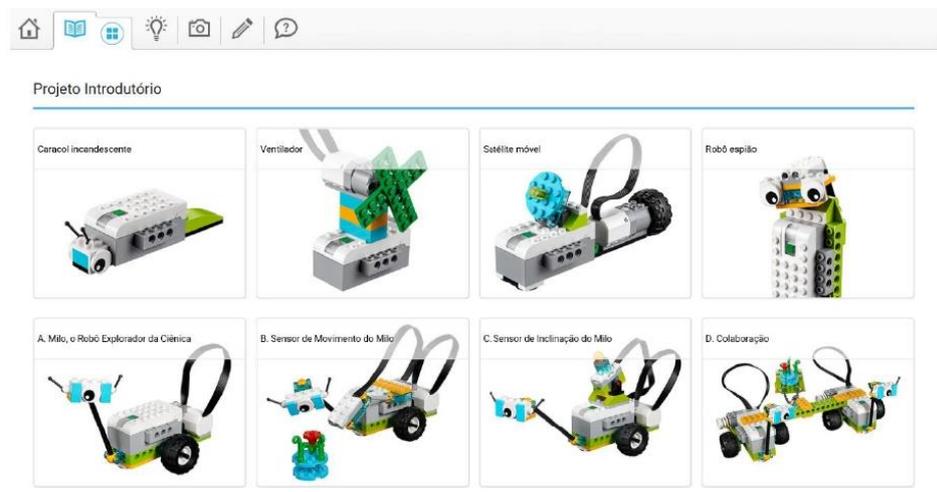
Procedimento inicial – Ligar o SmartHub ao software através de *bluetooth*

1. O trabalho a realizar, será em grupo de dois alunos. Cada grupo vai buscar um conjunto Lego Wedo 2.0 disponibilizado pela professora.
2. Nos dispositivos *bluetooth*, tenta emparelhar o *SmartHub* ao tablet ou computador, devem aparecer vários, no entanto apenas o número que está atribuído ao teu grupo deve ser selecionado. Caso não estejas a conseguir chama a tua professora.
3. Abre a aplicação do Wedo 2.0 e cria um projeto em branco (sinal +) em meus projetos.
4. Agora que já tens *SmartHub* a aparecer no software ou aplicação será boa prática, alterar o nome para o número que identifica o teu *SmartHub*. Deve estar identificado com um número.



Projeto A - Milo, o Robô Explorador da Ciência

1. Neste projeto a atividade proposta, passa por:
 - a. Veres um vídeo em conjunto e debater como explorar lugares remotos;
 - b. Construíres um robô explorador (Milo);
 - c. Programar o robô;
 - d. Partilhares as tuas ideias com os teus colegas.
2. Na aplicação do Wedo 2.0, procura o projeto introdutório A. Milo, o Robô Explorador de Ciência.



3. Segue todos os passos apresentados:
 - a. Ver o vídeo do Max e da Mia sobre o tema;
 - b. Responder à questão numa folha e entregar à professora;
 - c. Visualizar o vídeo "Usando as peças";
 - d. Construir o robô (Milo), usando a ajuda "comece a construir";
 - e. Programar o robô Milo com os blocos do motor.
4. Modifica o programa apresentado, com outras instruções (ex: quando parar emitir um som).
5. Caso tenhas dúvidas pede ajuda aos teus colegas ou à professora.



Projeto B - Sensor de Movimento do Milo

1. Nesta atividade vais trabalhar com sensores (sensor de movimento) que verifica se algum objeto está à sua frente. Este método será usado para o robô Milo encontrar uma planta e parar!
2. Na aplicação do Wedo 2.0, procura o projeto introdutório **B. Sensor de Movimento do Milo**.
3. Tarefas:
 - a. Ver vídeo "Usando as peças";
 - b. Adicionar as peças lego e o sensor de movimento ao robô Milo, de acordo com as instruções fornecidas;
 - c. Programar o robô com os blocos apresentados;
 - d. Verificar o funcionamento de cada bloco e compreender como funciona o sensor de movimento;
 - e. Partilhares as tuas ideias com os teus colegas.
4. Modifica o programa apresentado, com outras instruções (ex: inicialmente o robô está parado e inicia a sua marcha, após deteção de movimento no sensor).
5. Caso tenhas dúvidas pede ajuda aos teus colegas ou à professora.

Projeto C - Sensor de Inclinação do Milo



1. Nesta atividade vais trabalhar com sensores (sensor de inclinação) que verifica a posição do sensor relativamente ao plano. Este método será usado para o robô Milo para enviar mensagens!
2. Na aplicação do Wedo 2.0, procura o projeto introdutório **C. Sensor de Inclinação do Milo**.

3. Tarefas:

- a. Ver vídeo “Usando as peças”;
 - b. Adicionar as peças lego e o sensor de inclinação ao robô Milo, de acordo com as instruções fornecidas;
 - c. Programar o robô com os blocos apresentados;
 - d. Verificar o funcionamento de cada bloco e compreender como usar o sensor de inclinação e o bloco de repetição;
 - e. Partilhares as tuas ideias com os teus colegas.
4. Modifica o programa apresentado, com outras instruções (ex: inicialmente o robô está parado. Caso o sensor de inclinação estiver para cima, inicia a sua marcha se a inclinação for para baixo anda para trás).
5. Caso tenhas dúvidas pede ajuda aos teus colegas ou à professora.



Projeto D - Colaboração

1. Nesta atividade vais trabalhar em conjunto com outro grupo de trabalho. Serão necessários 2 *SmartHub*'s. Têm de existir um trabalho conjunto entre as duas equipas, por forma a controlar a direção dos robôs. Este método será usado para os dois Robôs transportarem uma planta!
2. Na aplicação do Wedo 2.0, procura o projeto introdutório **D. Colaborar**.
3. Tarefas:
 1. Escolhe a outra equipa;
 2. Ver vídeo “Usando as peças”;
 3. Adicionar as peças lego necessárias, de acordo com as instruções fornecidas;
 4. Programar o robô com os blocos apresentados;
 5. Verificar o funcionamento de cada bloco e compreender a sua função;
 6. Partilhares as tuas ideias com os teus colegas.
4. Modifica o programa apresentado, com outras instruções (ex: usar o sensor de inclinação para comandar os robôs, caso o sensor de inclinação estiver para cima, anda para a frente se a inclinação for para baixo para o motor).
5. Caso tenhas dúvidas pede ajuda aos teus colegas ou à professora.

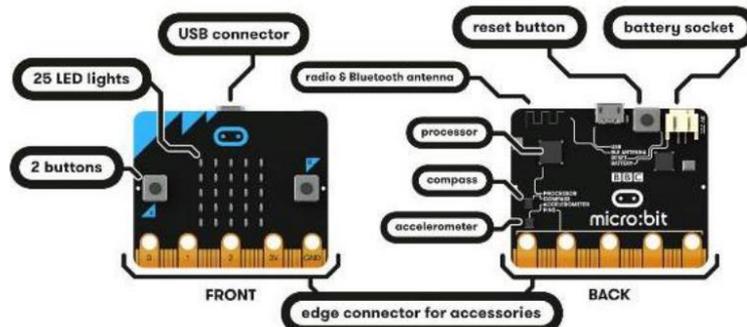


Bom trabalho 😊

Anexo E3 – Programa de intervenção – sessão 3 – Apresentação da placa Micro:Bit

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	ATIVIDADE 3
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)
	03/2022 - 50 min.
Ano Letivo 2021/2022	

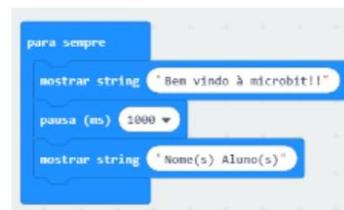
Atividade : Conhecer e explorar a placa Micro bit.



Hoje vamos dar os primeiros passos na descoberta da placa **microbit**, com ela poderemos realizar muitas experiências em diferentes áreas do saber.

Atividade 1

1. Crie uma pasta no seu ambiente de trabalho com o nome **micro_bit_F1**
2. Aceda ao endereço <https://makecode.microbit.org>
3. Clicar em **novo projeto**.
4. Alterar a linguagem para o idioma em **Português**, menu -> roda dentada  -> opção  **Idioma** e selecionar **Português**.
5. Alterar o nome do ficheiro, na parte inferior, para **F1_Atividade1_NomeAluno**.
6. Arrastar os blocos de forma a aparecer a mensagem "Bem vindo à microbit!!"
7. Insira outra mensagem com o seu nome.
8. Crie uma pausa de **1 s** entre as duas mensagens.
9. Alterar o código, para que no início de cada mensagem apareça uma imagem (*icon*) diferente.
10. Limpar o ecrã por um curto período entre as imagens.



11. Criar um *icon* e apresenta como a terceira imagem.
12. Guarda o exercício na pasta **micro_bit_F1** do ambiente de trabalho
13. Liga a placa micro:bit ao computador/tablet e transfere o programa para a placa.
Caso tenhas dificuldade, existe no final desta ficha instruções do procedimento, em caso de dúvidas chama a professora.



Atividade 2

14. Clicar em  **Início** e cria um projeto.
15. Altera o nome do ficheiro, na parte inferior, para **F1_Atividade2_NomeAluno**.
16. Na aba  **Entrada**, irá ter acesso a outras ferramentas como o controlo dos botões.
17. Experimente os seguintes blocos e verifique o seu funcionamento



18. Na mesma categoria adicionar o bloco agitar e adicionar uma imagem (*icon*).
Agora clicar na placa na nova opção 
19. No bloco de **arranque** programa para aparecer o seu nome.
20. Guarda o exercício na pasta **micro_bit_F1**.
21. Liga a placa micro:bit ao computador/tablet e transfere o programa para a placa.
22. Verifica o funcionamento dos botões A, B, A+B e o que acontece se agitares a placa.



Atividade 3

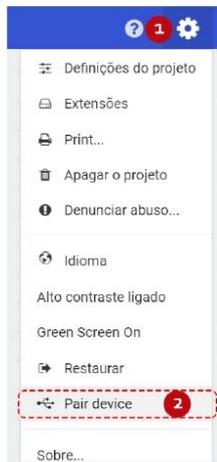
1. Clicar em  e criar um projeto.
2. Alterar o nome do ficheiro, na parte inferior, para `F1_Atividade3_NomeAluno`.
3. Programar o **botão A** para fazer um contador de 0 até 9. (Desenhar os números no bloco matriz)
4. Programar o **botão B** para fazer um contador em ordem decrescente de 9 até 0.
5. Programar o **bloco para sempre** de forma a animar um personagem (exemplo figura ao lado).
6. Guarda o exercício na pasta `micro_bit_F1`
7. Comprimir a pasta `micro_bit_F1` e enviar para a plataforma moodle em **Envio da F1**
8. Liga a placa micro:bit ao computador/tablet e transfere o programa para a placa.
9. Verifica o funcionamento dos botões A e B e o que acontece no arranque.



INSTRUÇÕES :Como transferir o programa para a placa micro:bit

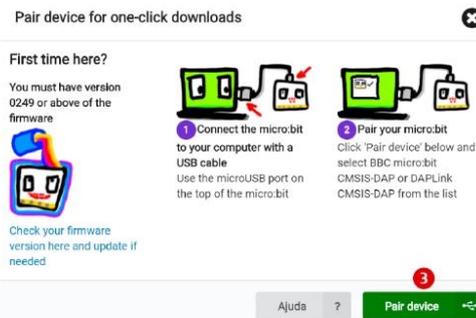
1. Existem duas formas de utilizar a placa Micro:Bit.
 - a. **Opção 1** – Instalar os drivers da placa no Windows;
 - b. **Opção 2** – Trabalhar como se a placa fosse uma *Pendrive*.
2. Em ambos os casos, necessitas em primeiro lugar de ligar a placa Micro:Bit ao computador com um cabo *USB*.
3. Sequência de passos relativos à **opção 1**:



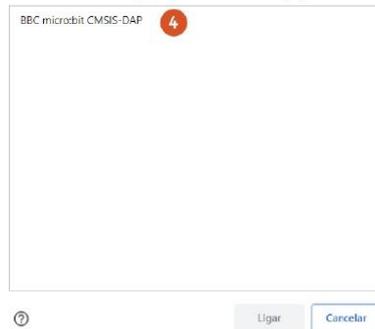


Ir a definições **1** e escolher a opção *Pair device* **2**

Na nova janela selecionar a opção *Pair device* **3**



makecode.microbit.org pretende estabelecer ligação



Escolher a placa **4** e clicar no botão **Ligar**.

Se tudo correr bem irás receber uma mensagem

Device paired! Try downloading now.

Agora já podes usar o botão

 **Transferir**

Sempre que necessites de enviar o código para a placa.

4. Sequencia de passos relativos à opção 2:

Ao ligares a placa ao computador, irá aparecer um novo drive no teu computador, com o nome de **MICROBIT**

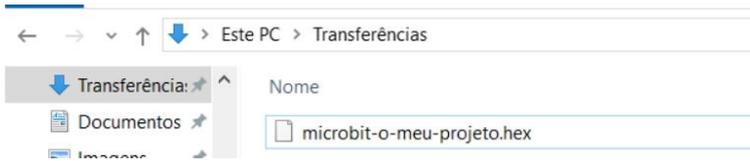


Agora clica no símbolo da disquete para gravar o teu programa para o computador.

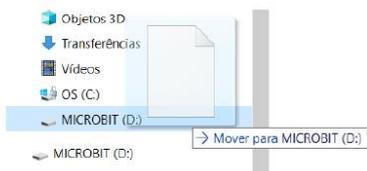
o meu projeto



Nesse momento é feito o *download* do programa para a pasta das transferências



Finalmente, basta arrastar o programa para a *drive* MICROBIT



Nesse preciso momento o LED da placa irá piscar, informando que está a receber o programa.

Nota: este segundo método é mais lento que o primeiro.

Bom trabalho!



	<p style="text-align: right;">ATIVIDADE 2</p> <p style="text-align: center;">OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)</p> <p style="text-align: right;">xx/02/2022 - 50 min.</p> <p style="text-align: right;">Ano Letivo 2021/2022</p>
---	---

Objetivos: Nesta ficha de trabalho, vais aprender o que são variáveis e para que servem. Para isso, vais trabalhar com a placa micro:bit e explorar alguns exercícios.

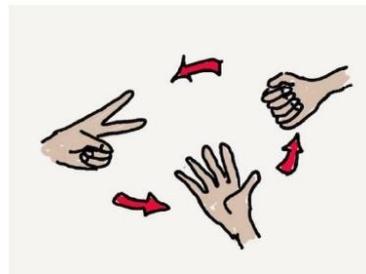


Parte 1 - Atividade sem computadores e programação (unplugged)

Registo de pontuação de um Jogo

Objetivo

Nesta atividade, vais jogar o jogo “pedra, papel, tesoura” e registar numa folha os resultados.



Instruções

- 1) Grupos de três alunos

Dois alunos jogam e o terceiro regista numa folha os resultados do jogo.

- 2) Registo dos resultados do jogo

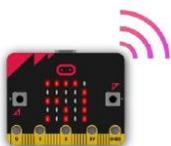
O aluno responsável deve registar os seguintes tópicos, durante 1 minuto de jogo, em grupos de 4 rodadas:

- Registrar o nome dos colegas;
- Quantas vezes o colega A ganha;
- Quantas vezes o colega B ganha;
- Quantas vezes empatam;
- Quantas rodadas jogaram



Quando terminar, somar as pontuações e quantas rodadas jogaram.

Voltar a jogar novamente, por mais um minuto (o aluno que regista agora vai jogar com quem ganhou e o aluno que perdeu passa a registar os resultados). Quando terminar, somem suas pontuações e registem quantas rodadas jogaram.



Discussão

Compartilhar os resultados obtidos

No vosso registo das pontuações, o que acham que é constante, ou seja, valores que não mudam ao longo de uma sessão de jogo.

E o que podem ser variáveis, valores que mudam ao longo de uma sessão de jogo.

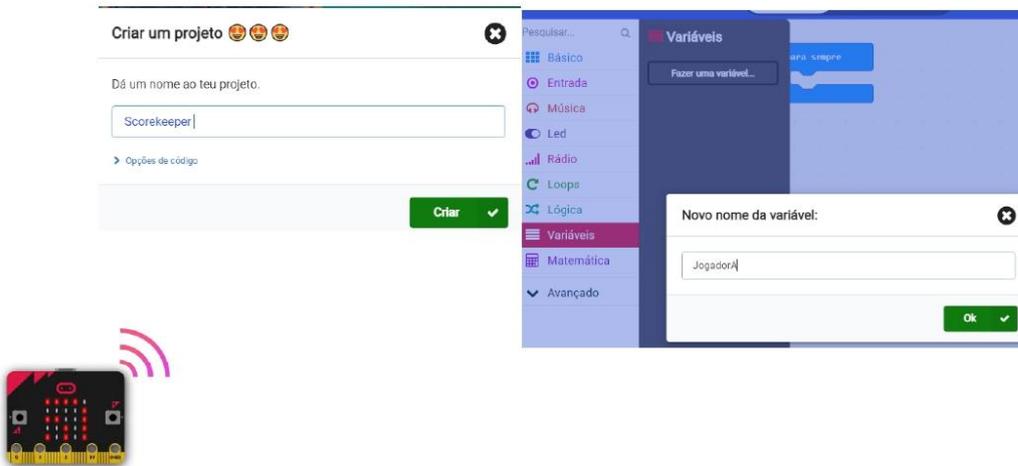
E de que tipo podem ser essas variáveis que identificaram.

Parte 2 - Atividade com o micro:bit – Desenvolver a aplicação para registar os resultados do jogo.

Como foi discutido e observado, vamos necessitar de três variáveis:

- O número de vezes que o primeiro jogador ganha
- O número de vezes que o segundo jogador ganha
- O número de vezes que os jogadores empatam

- 1) No MakeCode, inicia um novo projeto e nomeia-o como: **Scorekeeper** . No menu Variáveis, cria e identifica essas três variáveis: **JogadorA** , **JogadorB** , **Empates**.



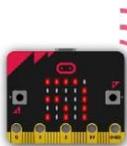
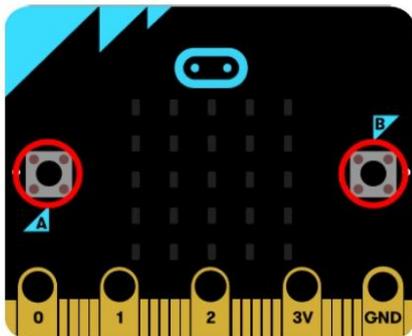
Inicializando o valor da variável

- 2) É importante dar às tuas variáveis um valor inicial. O valor inicial é o valor que a variável manterá cada vez que o programa for iniciado. Para nosso programa de contador, daremos a cada variável o valor 0 (zero) no início do programa.



Atualizando o valor da variável

- 3) No programa, queremos registrar o número de vezes que cada jogador ganha e o número de vezes que eles empatam. Vamos usar os botões A e B no micro:bit para fazer isso.



- O pseudocódigo, escrito em palavras, é:
- Pressiona o botão A para registrar uma vitória para o Jogador A
- Pressiona o botão B para registrar uma vitória para o Jogador B
- Pressiona o botão A e o botão B juntos para registrar um empate

Já inicializamos as variáveis, agora vamos programar para atualizar os valores a cada rodada do jogo.

- Ao carregamos no botão A, registamos uma vitória para o Jogador A. Assim temos de adicionar mais um valor ao valor atual da variável do JogadorA.
- Ao carregamos no botão A, registamos uma vitória para o Jogador A. Assim temos de adicionar mais um valor ao valor atual da variável do JogadorB.
- Ao pressionar o botão A e o botão B, ao mesmo tempo, para registar um empate. Assim temos de adicionar mais um valor ao valor atual da variável do Empate.



Relembrar que, ao iniciar o jogo, todas as variáveis foram inicializadas com o valor igual a zero.



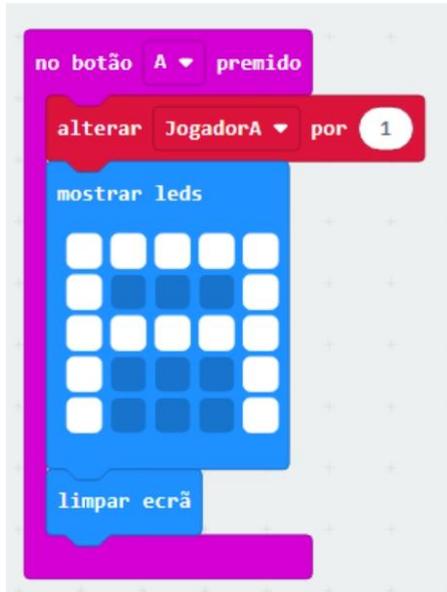
Adicionando feedback ao utilizador

Sempre que o aluno pressionar o botão A, o botão B ou ambos os botões juntos, vais ter que dar ao utilizador um feedback visual reconhecendo que o utilizador pressionou um botão:

- Um 'A' cada vez que o utilizador pressiona o botão A, para registar uma vitória para o Jogador A,
- Um 'B' para cada vez que o utilizador pressiona o botão B, para registar uma vitória para o Jogador B,
- Um 'E' para cada vez que o utilizador pressiona o botão A + botão B juntos, para registar um empate.
- Podemos exibir um 'A', 'B' ou 'E' utilizando o bloco 'mostrar leds' ou o bloco 'mostrar string'.



Neste exemplo, temos o exemplo com bloco 'mostrar leds' apenas para o Botão A. Deves completar o código para os restantes botões (B e A+B).



Observa que adicionamos o bloco 'limpar ecrã', depois de mostrar 'A', 'B' ou 'E'. O que acontece se não usares este comando? Verifica!

Mostrando os valores finais das variáveis

Para finalizar o programa, vamos adicionar um código que diga ao micro:bit para mostrar os valores finais das nossas variáveis. Como já usamos os botões A e B, podemos usar o bloco 'em agitar'. Podemos usar os blocos 'mostrar string', 'mostrar leds', 'pausa' e 'mostrar número' para



exibir esses valores finais de maneira clara.

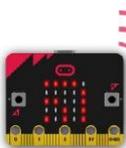


Quando agitares o micro:bit, ele vai mostrar os resultados do JOGO. Na figura anterior vai mostrar a palavra/string Resultados, depois mostra a letra A (que é a letra do jogador A) e o valor das vitórias desse jogador (variável **JogadorA**) e esperamos 1000 ms ou seja 1 segundo.

Agora, vais seguir os mesmos passos para mostrar o número de vitórias do Jogador B. Vê se consegues:

- Adicionar o mostrar leds com a letra B
- Adicionar o bloco mostrar número
- Da caixa de ferramentas de variáveis, arraste a variável 'JogadorB' para substituir o 0 no bloco 'mostrar número'.
- Adicionar o bloco pausa para 1 segundo (1000 ms).

Continua o código para mostrar o número de empates:



- Adicionar o mostrar leds com a letra E
- Adicionar o bloco mostrar número
- Da caixa de ferramentas de variáveis, arraste a variável 'Empates' para substituir o 0 no bloco 'mostrar número'
- Adicionar o bloco pausa para 1 segundo (1000 ms).

Há um último bloco para adicionar. Consegues adivinhar qual é? Da Caixa de ferramentas básica, arrasta um bloco 'limpar ecrã' para baixo do bloco 'pausa'.

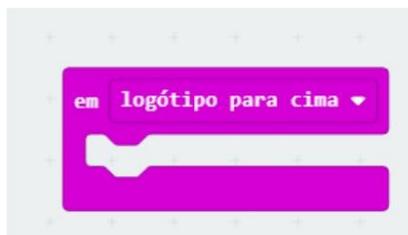


Acrescentar o número de jogadas

Queremos mostrar o número de jogadas total. Como podemos alcançar esse número? Para fazer isso, temos de adicionar os valores armazenados nas variáveis que foram criadas para cada jogador ganhou e quantas vezes empatou. Insere o bloco logótipo para cima



- Primeiro, mostra um texto a dizer 'Total de Jogadas' para mostrar o número total de rodadas jogadas. Nosso programa adicionará os valores armazenados nas variáveis JogadorA, JogadorB e Empates e mostra a soma desses valores



- Os blocos para as operações matemáticas somar, subtrair, multiplicar e dividir estão listados na seção Matemática da Caixa de ferramentas.



Guarda o teu trabalho, envia para a placa micro:bit e testa o funcionamento. Caso não esteja correto tenta descobrir o erro e volta a testar até chegares à solução!



Anexo E5 – Programa de intervenção – sessão 5 – Testar condições com a placa Micro:Bit

	ATIVIDADE 5
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)
	03/2022 - 50 min.
	Ano Letivo 2021/2022

Objetivos: Nesta ficha de trabalho, vais aprender o que são condições/condicionais e para que servem. Para isso, vais trabalhar com a placa micro:bit e explorar alguns exercícios.

Como já deves saber, os programas de computador são instruções que dizem ao computador como processar e devolver resultados. Uma parte muito importante de programar é dizer ao computador quando este deve efetuar certas instruções. Para tornar isto possível, existe os **condicionais**, que executam certas instruções quando certas condições/regras são cumpridas. Um exemplo do dia-a-dia seria quando os vossos pais dizem “**Se não limpares o teu quarto, não podes jogar no teu computador**”. Isto é um condicional, pois se vocês cumprirem a condição de limparem o vosso quarto, podem jogar.



Conseguem pensar em mais exemplos de condicionais?

As condicionais seguem o formato **SE** isso, **ENTÃO** aquilo.

SE (condição for atendida), **ENTÃO** (ação realizada)

SE (condição for atendida), **ENTÃO** (ação realizada), **SENÃO** (ação diferente realizada)

SE estiver a nevar, **ENTÃO** uso botas, **SENÃO** uso sapatilhas.

Parte 1 - Atividade sem computadores e programação (unplugged)

Treino de condições

Aleatoriamente alguns alunos serão alinhados numa extremidade da sala de aula com o objetivo de alcançar o outro lado da sala de aula. Pretende-se “chamar” os alunos mediante instruções usando diferentes condições. Os alunos visados avançarão ou não, dependendo da declaração condicional específica.

Exemplo:

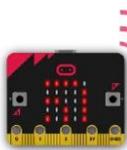
SE o aluno tem vestindo algo verde, **ENTÃO** dá um passo à frente.

SE o aluno tem a letra 'e' no seu primeiro nome, **ENTÃO** dá 3 passos à frente.

SE o aluno estiver de sapatilhas, **ENTÃO** dá um passo à frente, **SENÃO** dá 2 passos à frente.

SE o aluno faz anos este mês, **ENTÃO** dá um salto de gigante para a frente.

Outros exemplos, podem ser aplicados.



Depois dos alunos entenderem o conceito e ideia do jogo, eles próprios poderão inventar outras condicionais (que atendem à aprovação do professor).

Caso os alunos tenham entendido o conceito, podem ser adicionados os operadores relacionais **E** e **OU** as condições.

Exemplo:

SE o aluno tem cabelo castanho E olhos castanhos ENTÃO, dá um passo à frente.

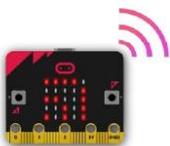
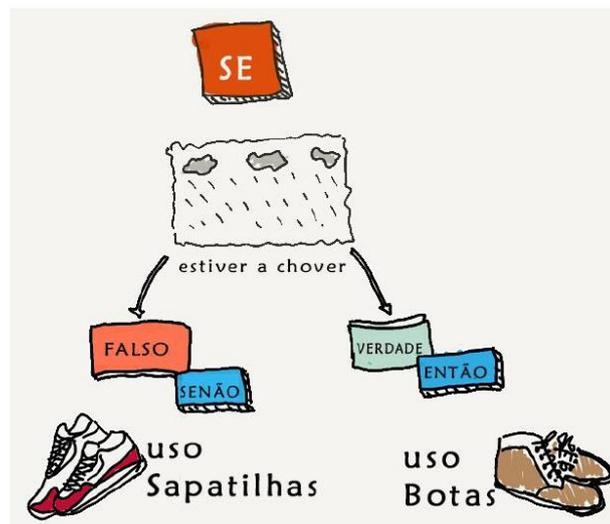
O próximo passo é treinar as condições encadeadas...

Exemplo de um **SE** dentro de outro **SE**

Exemplo:

SE o aluno tem calçadas sapatilhas, ENTÃO SE usa meias brancas, ENTÃO dá três passos à frente.

Permitir que os alunos criem as suas próprias estruturas condicionais e apresentem à turma os seus exemplos.



Parte 2 - Atividade com a placa micro:bit

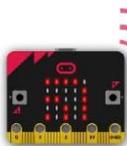
Vamos agora fazer um pequeno exercício com condicionais! O objetivo é mostrar *emojis* diferentes quando carregas no **botão A** e no **botão B**. Vamos começar!



Primeiro, acede ao link <https://makecode.microbit.org/> e adiciona o bloco **para sempre**. Dentro desse bloco temos de adicionar duas condições: **se o botão A for premido** e **se o botão B for premido**. Para isso, usamos os blocos **se ... então** com a condição **botão A está premido** e **botão B está premido**.



Por último, vamos mostrar os *emojis*. Dentro de cada condição coloca o *emoji* que quiseres, que se encontra na secção **Básico**. No final, deve ficar algo similar à imagem seguinte:



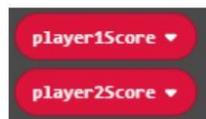
Como procederias para introduzir mais uma condição, se o **botão A+B for premido**? Deves incluir mais uma condição no sinal  e atribuir um *icon* diferente.

Para finalizar, deves incluir um *icon* que deve aparecer se não for carregado nenhum botão!

Parte 3 – Jogo cara ou coroa

1. Clicar em  e cria um projeto.
2. Alterar o nome do ficheiro, na parte inferior, para **Jogo-NomeAluno**.
3. Vamos criar um pequeno jogo. O **jogo da cara ou coroa** para dois jogadores.
4. Inicialmente no arranque vamos informar os jogadores, para isso codificar o arranque de acordo com a figura ao lado.

5. O próximo passo é criar duas variáveis para cada um dos jogadores



6. O botão A vai ser para o jogador1 e o botão B para o jogador2. Se o jogador 1 carregar no botão A poderá sair ou um coração ou uma cruz. Caso saia o coração o jogador recebe 1 ponto, se sair o x o jogador1 não recebe pontos. Exemplificado na imagem do lado.



7. Programar o botão B para o jogador2, que é idêntico ao código do jogador1.

8. Criar um código necessário para fazer o *reset* (limpar as variáveis) do jogo de forma a jogar de novo. Irá mostrar a mensagem de RESET, fazer uma pausa de 1 segundo e limpa o ecrã. Esta ação será originada sempre que viramos a placa com o logotipo para baixo, como exemplificado na figura ao lado.

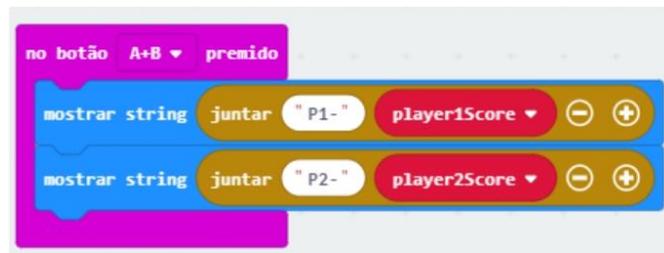


```

em logotipo para baixo
  definir player2Score para 0
  definir player1Score para 0
  mostrar string "Reset"
  pausa (ms) 1000
  limpar ecrã
  
```



9. Por fim mostrar a pontuação de cada jogador, recorrendo às mensagens e a junção de mensagens. A pontuação será mostrada quando carregarem simultaneamente no botão A e no botão B. Como indicado na imagem que se segue.



```

no botão A+B premido
  mostrar string juntar "P1-" player1Score
  mostrar string juntar "P2-" player2Score
  
```

10. Testar o programa e observar o resultado.

11. Como desafio tente acrescentar ao bloco anterior, o código necessário para apresentar uma mensagem do nome do jogador que tem a pontuação mais alta.

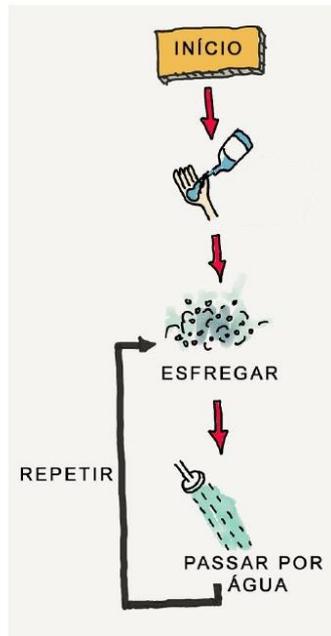
12. Testar o programa



Anexo E6 – Programa de intervenção – sessão 6 – O que são iterações?

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	ATIVIDADE 6
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)
	Maio de 2022
Ano Letivo 2021/2022	

Objetivos: Nesta ficha de trabalho vais aprender o que são iterações/repetições e para que servem. Para isso, vais trabalhar com a placa micro:bit e explorar alguns exercícios.



Em programação, iteração é a repetição de uma sequência de código. Um ciclo é uma forma de iteração onde uma sequência de código é repetida até uma certa condição ser satisfeita.

Por exemplo, imaginem que estão a lavar o cabelo. Colocam o shampoo e passam por água. Se o vosso cabelo ainda tiver shampoo, têm de passar água novamente. Este ciclo continua até terem o cabelo livre de shampoo.

Na programação de computadores, a iteração é a repetição de uma sequência de comandos. Os programadores podem usar um tipo especial de código chamado loop (ciclo), de forma a englobar outros comandos que desejam repetir como uma forma de iteração. Um loop repete o código até que uma determinada condição seja verificada.

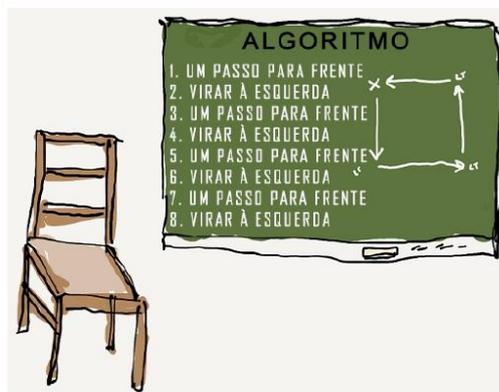


Parte 1 - Atividade sem computadores e programação (unplugged)

Andar um quadrado

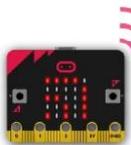
A figura representa o algoritmo para desenhar um quadrado em oito passos.

Em grupo de dois alunos, colocar uma cadeira ao centro e, a um dos alunos solicitar que diga as instruções para que o colega desenhe um quadrado em torno da cadeira.



Questionar aos alunos sobre:

1. Se existe necessidade de andar mais que um passo.



2. Questionar se o algoritmo funciona.
3. Questionar que instruções são repetidas.
4. Verificar quantas vezes são repetidas.
5. Então como poderíamos reescrever o código?
6. Apresentar o algoritmo apenas em três linhas.



No final, os alunos acabaram de reescrever o código de oito linhas, em apenas três linhas de código, usando um ciclo/loop. O comando "repeat", cria um ciclo (loop). O código dentro do ciclo é repetido um certo número de vezes até que uma dada condição se verifique. A condição neste algoritmo é que o código dentro do ciclo seja repetido 4 vezes. Assim que chega a essa condição o quadrado está concluído.

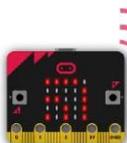
Esta é uma boa oportunidade para que os alunos pensem nos benefícios de ter menos linhas de código. Algumas razões possíveis (menos digitalização, economia de tempo, menos probabilidades de errar, mais fácil de ler o código, menos linhas de código para processar...)

Desafio: Qual seria o código para criar um quadrado no sentido dos ponteiros do relógio (sentido horário). Comparar os dois códigos. Qual seria a alteração?

Parte 2 - Codificação com loops (ciclos)

Para as atividades de codificação de hoje, vais usar três blocos de ciclos diferentes no Microsoft MakeCode:

- bloco 'repeat' / repetir - este bloco repete o código (n) vezes.
- Bloco 'while' / enquanto - este bloco executa o código enquanto a condição dentro dele for verdadeira.
- Bloco 'for' / para - este bloco repete o código (n) vezes, mas com uma variável.



Vamos criar **três atividades de codificação** para demonstrar como funciona cada tipo de bloco de repetição.

Atividade de codificação 1: Codifique um sprite (ponto luminoso do micro:bit) para percorrer um quadrado com o loop 'repetir'

Para começar, vamos relembrar o código inicial da cadeira

- Um passo para à frente
- Virar à esquerda
- Um passo para à frente
- Virar à esquerda
- Um passo para à frente
- Virar à esquerda
- Um passo para à frente
- Virar à esquerda



Apenas neste caso, vais codificar um "sprite" ou uma luz de LED para movimentar no micro:bit. No Microsoft MakeCode, inicia um novo projeto e nomeia-o como: **quadrado andante**. Podes deixar o bloco "no arranque" na área de trabalho de codificação, mas podes eliminar o bloco de "para sempre"



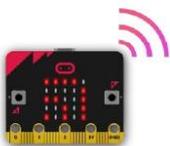
Criar uma variável Sprite

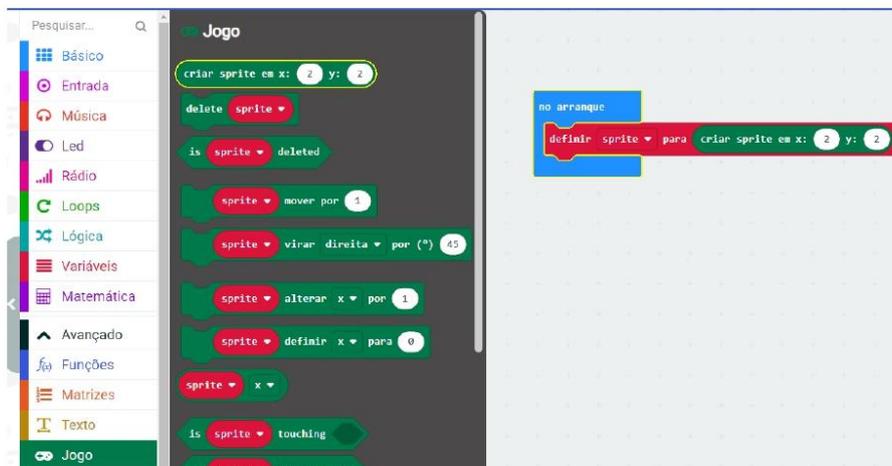
No início, vamos querer que o sprite apareça. Para que isso aconteça, necessitamos de uma variável com o nome *sprite*.



Cria um Sprite

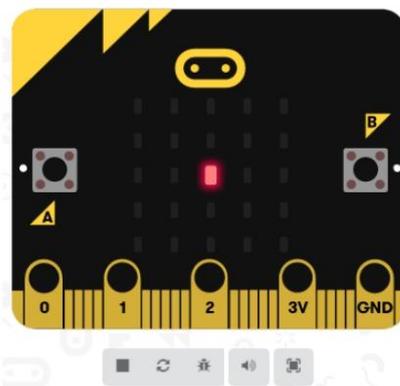
Clica na categoria *Avançado* na parte inferior da Caixa de ferramentas  Avançado. Isso abrirá mais do menu de ferramentas. Clica na categoria *Jogo*  Jogo e arrasta um bloco oval 'criar sprite' para a área de trabalho de codificação. Solta-o no bloco definir sprite, substituindo 0 por 2.





Agora já deves ver o 'sprite', ou uma luz LED vermelha aparecer no meio do simulador micro:bit.

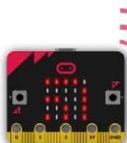
Na face do micro:bit há uma grelha de 5 x 5 luzes LED. As coordenadas X são as posições de luz horizontais que vão de 0 a 4, e as coordenadas Y são as posições de luz verticais que vão de 0 a 4 também. Pode-se ver nos blocos de código que criamos nosso sprite na posição central X, Y (2, 2). Se quisermos iniciar nosso sprite no canto superior esquerdo da tela, teremos de alterar as coordenadas iniciais para x=0 e y=0 (0, 0).



Movendo o sprite

Agora vamos fazer o mover o nosso sprite pela face do micro:bit. Isso será feito quando pressionar um botão.

Na gaveta da Caixa de ferramentas de **entrada**, arrasta um bloco **no botão A premido** para a área de trabalho.



Vamos necessitar de mais dois blocos no menu JOGO. Se consultares o algoritmo, podemos encontrar os blocos que vais necessitar, ou seja, mover o sprite para frente e girar o sprite.

Para fazer o sprite mover de um lado da tela para o outro (como se estivesse a andar a volta da cadeira), vais precisar de mover o sprite 4 lugares. Portanto, altere o valor no bloco mover por' de 1 para 4 .

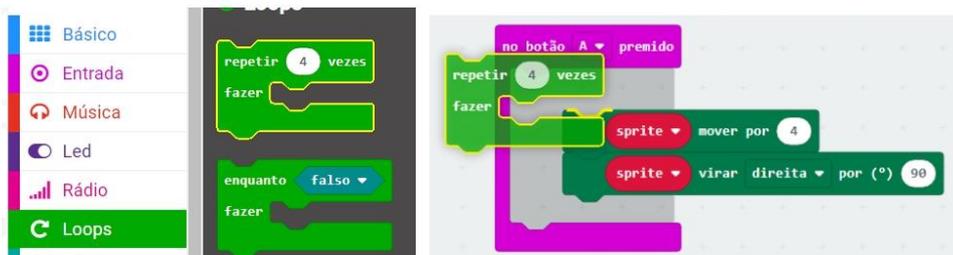


Para fazer o sprite girar para fazer um quadrado, temos de girar/virar para um lado (esquerda/direita) o 'virar à direita por 90 graus. Por enquanto, não há problema em deixar o sprite virando para a direita em vez da esquerda.

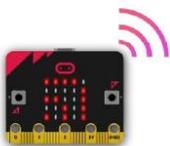
O bloco 'repetir'

Seguindo nosso algoritmo, agora podemos adicionar mais três blocos Mover e Girar para fazer nosso sprite andar num quadrado, mas existe uma maneira mais fácil e eficiente de codificar isso! Usando um loop/ciclo de repetição.

Clica na categoria Loops na caixa de ferramentas. Arrasta o bloco de repetição para a área de trabalho de codificação e coloca-o ao redor dos blocos 'mover por' e 'virar' do sprite.



Observa que o bloco 'repetir' contém um valor padrão de 4. Isso significa que ele repetirá quaisquer blocos de código que contenha quatro vezes (4x).

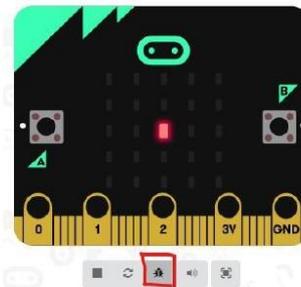


Segue em frente e executa o programa. Verifica o que vai acontecer ao sprite pressionando o botão A no simulador.

O que aconteceu? Viram o movimento do sprite? Não? Porquê? Porque acontece tão rápido, e não conseguimos ver passo a passo apenas o resultado.

Usar câmara lenta do MakeCode

Um recurso útil do Microsoft MakeCode é o modo depuração ou câmara lenta. Clica no ícone de 'bug' em baixo do simulador micro:bit. Isso retardará a execução do programa e destacará partes do código para que visualizes passo a passo qual linha de código está a ser processada. Esta tarefa chama-se depuração.

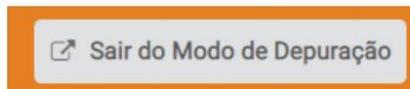


Na nova janela que aparece, podes verificar que o símbolo do caracol está ativo, o que quer dizer que esta em câmara lenta e podemos ver o funcionamento passo a passo.

Agora executa o teu programa na seta verde (▶). Consegues ver as diferentes linhas do teu código destacadas enquanto o programa é executado? Consegues ver o movimento do sprite?



Para sair deste modo de depuração apenas tens de clicar no botão que está no canto superior do lado direito.



Adicionar uma pausa

Neste momento, o código está feito para o sprite se mova sem parar e execute o código até ao fim das instruções! Às vezes, esquecemos o quão rápido os computadores são. Para que consigas ver o movimento do sprite mesmo no modo "normal", vamos adicionar uma pausa ao nosso programa logo após cada movimento do sprite. Isso dará aos nossos olhos humanos a oportunidade de vê-lo a mover.



Adiciona uma pausa de 200 milissegundo de acordo com a imagem

```
no botão A ▾ premido
  repetir 4 vezes
    fazer
      sprite ▾ mover por 4
      sprite ▾ virar direita ▾ por (°) 90
      pausa (ms) 200 ▾
```

Executa o programa no micro:bit. Agora podemos ver o movimento do sprite. Ele ainda está a mover muito rápido, mas pelo menos podemos vê-lo a mover...

Se tiveres tempo, experimenta alterar o valor de Pausa, o número de vezes para Repetir ou o número de espaços para mover o Sprite, verifica se essas mudanças afetam o programa.

O próximo desafio é encontrar uma solução para que ao mover o sprite de posição, faz uma pausa de 200 milissegundo (desta forma podemos ver o rasto de deslocamento).

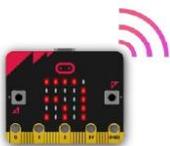


```
no botão A ▾ premido
repetir 4 vezes
  fazer
    sprite ▾ mover por 1
    pausa (ms) 200 ▾
    sprite ▾ mover por 1
    pausa (ms) 200 ▾
    sprite ▾ mover por 1
    pausa (ms) 200 ▾
    sprite ▾ mover por 1
    pausa (ms) 200 ▾
    sprite ▾ virar direita ▾ por (°) 90
```



Testa o código várias vezes. Como podes simplificar este código? Apresenta uma solução.

Os alunos vão testar dois ciclos encadeados. O ciclo interior faz o deslocamento de 4 posições e o exterior repete essa ação 4x.



Anexo E7 – Programa de intervenção – sessão 7 – Mini-Projeto “Hot Potato”

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	ATIVIDADE 7
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR) 05/2022 - 50 min. Ano Letivo 2021/2022

Objetivo:

Criar um jogo básico *hot potato*, ou seja, o jogo da batata quente!

Material necessário:

- Uma batata
- Fita cola
- A placa micro:bit
- Pilhas



Como já deves saber o jogo da batata quente é um jogo tradicional, em que os jogadores se colocam em roda (de pé ou sentados) para ver quem consegue passar um objeto o mais rapidamente possível, ao som de uma lengalenga sobre a “batata quente”. Quem ficar com o objeto na mão no fim da lengalenga sai. O jogo termina quando todos forem eliminados, e restar apenas o vencedor.

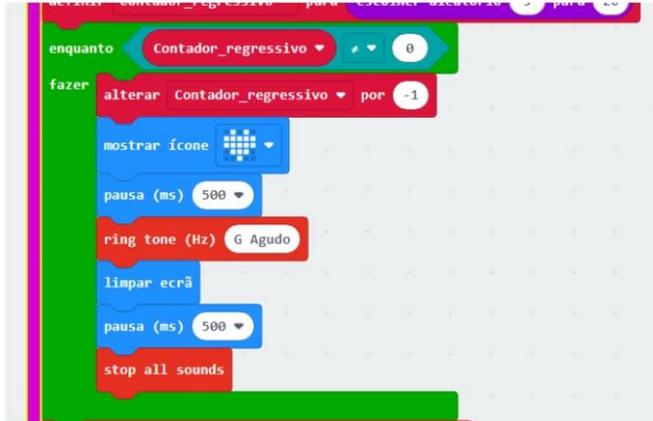
A tarefa de hoje é reproduzir este jogo para a placa micro:bit, para isso vai ser necessário fazer um programa que seja capaz de executar a tarefa anterior.

1. Iniciamos o jogo com a informação de início “S” de *Start*.
2. Vamos iniciar o jogo e a contagem de tempo ao carregar no botão A da placa micro:bit. O tempo de jogo sem “explodir” a batata vai ser definido de forma aleatória um número entre 5 e 20. E o número sorteado será guardado numa variável designada por `contador_regressivo`.

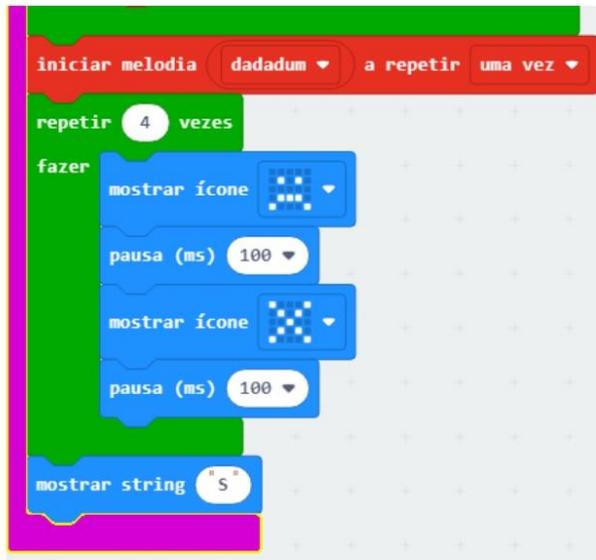


3. Vamos usar um ciclo **enquanto...fazer** que vai decrementar a variável de uma unidade, mostrar a animação de um coração e tocar um som. Este procedimento será sempre efetuado enquanto a variável `contador_regressivo` for diferente de zero.






4. Quando a variável contador for igual a zero, deixamos o ciclo e informamos o jogador que tem a batata na mão que perdeu, através de uma melodia.
5. Como podes observar na figura ao lado também temos um ciclo que executa 4x um símbolo triste e um X a informar que o jogador perdeu!
6. No fim informa que estamos prontos para jogar novamente.
7. Verifica o funcionamento do programa.




8. A próxima tarefa é enviar o programa para a placa micro:bit e acoplar as pilhas de forma a fornecer energia à placa.
9. Vamos agora pegar na batata e com fita-cola seguramos à placa micro:bit à batata.
10. Coloca 5 cadeiras na sala e joga o jogo com 5 colegas teus!

	<p style="text-align: right;">ATIVIDADE 7</p> <p style="text-align: center;">OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)</p> <p style="text-align: right;">05/2022 - 50 min.</p> <hr/> <p style="text-align: right;">Ano Letivo 2021/2022</p>
---	--

11. Agora que já entendeste como se joga e como está feito o programa, existem muitas situações que podes adaptar/alterar no programa. Por exemplo
- a. Alterar/aumentar o tempo de jogo, alterando o intervalo de valores de forma aleatória;
 - b. Alterar o tipo de sons emitidos;
 - c. Alterar as animações;
 - d. Ou outras sugestões tuas.



12. Explora e modifica o código

😊 Bom trabalho!



Anexo E8 – Programa de intervenção – sessão 8 – Mini-Projeto “Buzzer Game”

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	ATIVIDADE 8
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR) 05/2022 - 50 min. Ano Letivo 2021/2022

Objetivo:

Criar o jogo do arame, o *Buzzer Game*!

Material necessário:

- Plasticina, tesoura e fita-cola adesiva
- Fio Condutor
- Fios tipo crocodilo
- A placa micro:bit V2
- Pilhas



Se questionares os teus pais sobre o jogo do arame ou o jogo em que não se pode tocar num fio que emite um som ou acende uma luz, provavelmente eles irão reconhecer e descrever como funciona e contar como era divertido!

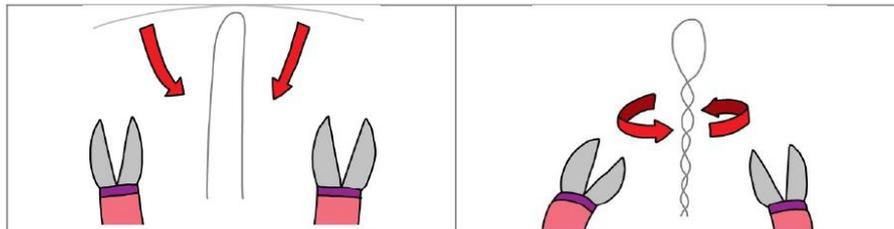
De forma a entenderes como funciona, visualiza este pequeno vídeo do Mr. Bean com os teus colegas e professor:

<https://www.youtube.com/watch?v=bRpUauseTVw>

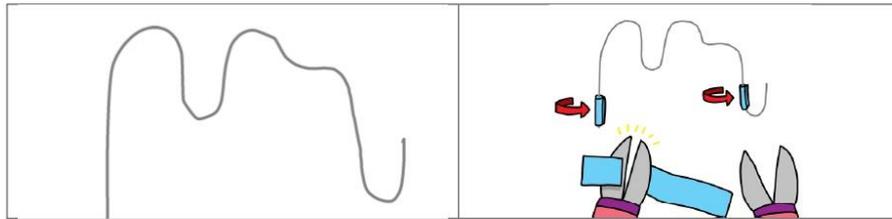
A tarefa de hoje é reproduzir este jogo para a placa micro:bit, para isso vai ser necessário construir fisicamente o suporte do jogo e depois a codificação.

Primeira Tarefa - construir o suporte físico

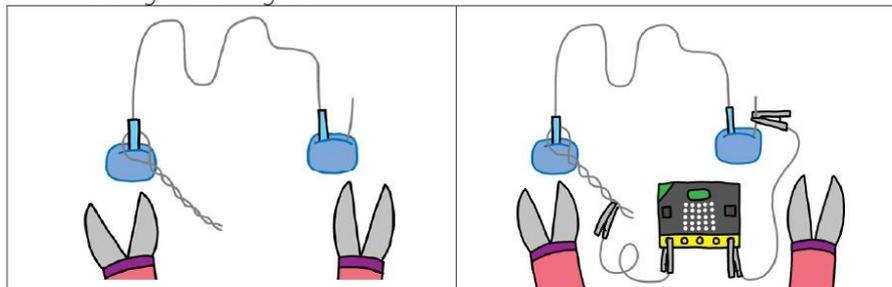
1. Verifica que tens o material necessário, e o primeiro passo é fazer varinha. Pede ao professor o pouco de fio condutor com 20 cm de comprimento e dobra-o ao meio, de forma a criar um orifício na parte superior. De forma, a dar mais resistência à varinha, podes torcer os dois pedaços de fio juntos.



2. Para fazer o percurso, solicita outro pedaço de fio condutor com cerca de 30 cm de comprimento e dobra de forma a criares um caminho por onde irá circular a ponta da varinha. Nota: não deves fazer um percurso difícil, nesta fase. Nas extremidades isola com fita-cola com cerca de 4 cm do fio condutor (ver imagens seguintes).



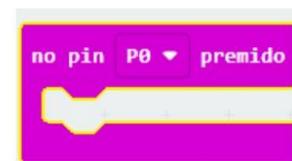
3. Insere a varinha dentro do percurso curso, cria uma base com plasticina para segurar o percurso. Noma das extremidades deixa visível 2 cm de fio condutor, depois com os fios crocodilo liga aos pins da placa micro:bit, como exemplificado nas seguintes imagens.



4. O pino (GND) vai ligar a uma das extremidades do percurso (pode ser à extremidade saliente) e o PIN 0 liga a extremidade da varinha. Neste momento a parte física está preparada, o próximo passo é fazer a programação do jogo.

Segunda Tarefa – Desenvolver a codificação

5. Abre o editor do micro:bit, o *makecode* e cria um novo projeto como nome **Buzzer_Game**.
6. Apaga os blocos iniciais (**no arranque** e **para sempre**)
7. Cria uma variável, designada por **FALHAS**, que irá conter o número de toques no fio.
8. Adiciona o bloco para ao carregar no **botão A** o valor da variável seja 0.
9. Insere o bloco que controla a entrada dos pinos da placa micro:bit, neste caso o **PINO** onde está ligada a varinha.
10. Dentro do bloco insere um ícone com um X
11. Adiciona o bloco com espera de 1s



12. Adiciona o bloco que se tocarem no fio com a varinha, adiciona um valor à variável **FALHAS**

13. Adiciona o código necessário, para ao premir o **botão B** mostre o número de toques no fio.



14. O teu código deve estar semelhante ao da figura seguinte.



15. Envia o código para a placa micro:bit e alimenta-a com as pilhas.

16. Testa o jogo e partilha com os teus colegas

17. Agora que já entendeste como se joga e como está feito o programa, existem muitas situações que podes adaptar/alterar o programa e o suporte físico. Por exemplo

- a. Criar um percurso mais longo;
- b. Alterar o orifício da varinha para ficar mais pequeno;
- c. Adicionar mais curvas no percurso;
- d. Inserir um som ao tocar no fio;
- e. Adicionar um LED vermelho que acende ao tocar no fio;
- f. Alterar as animações;
- g. Fazer um contador regressivo, apenas com 9 toques;
- h. Ou outras sugestões tuas.

18. Explora e modifica o código

😊 Bom trabalho!



Anexo E9 – Programa de intervenção – sessão 9 – Apresentação do Robô Ozobot, programação por cores

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p>  <p>CIÊNCIAS DA COMPUTAÇÃO</p>	<p>ATIVIDADE 9</p> <p>OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)</p> <p>06/2022 - 50 min.</p>
	<p>Ano Letivo 2021/2022</p>

Objetivo:

Conhecer o robô Ozobot

Programar o robô Ozobot através de cores

Material necessário:

- Canetas de cores (Kit Ozobot)
- Robô Ozobot
- Tabela de cores e ações



Hoje vais conhecer um robô muito pequeno com grandes potencialidades. Na primeira parte da aula vais identificar as diferentes partes constituintes do robô, como os sensores e motores. Também vais saber como o calibrar.

Primeira parte - Conhecer o robô Ozobot

Nesta parte vais visualizar alguns vídeos curtos, sobre a constituição do robô ozobot. Está atento e se tiveres dúvidas solicita esclarecimento à professora.

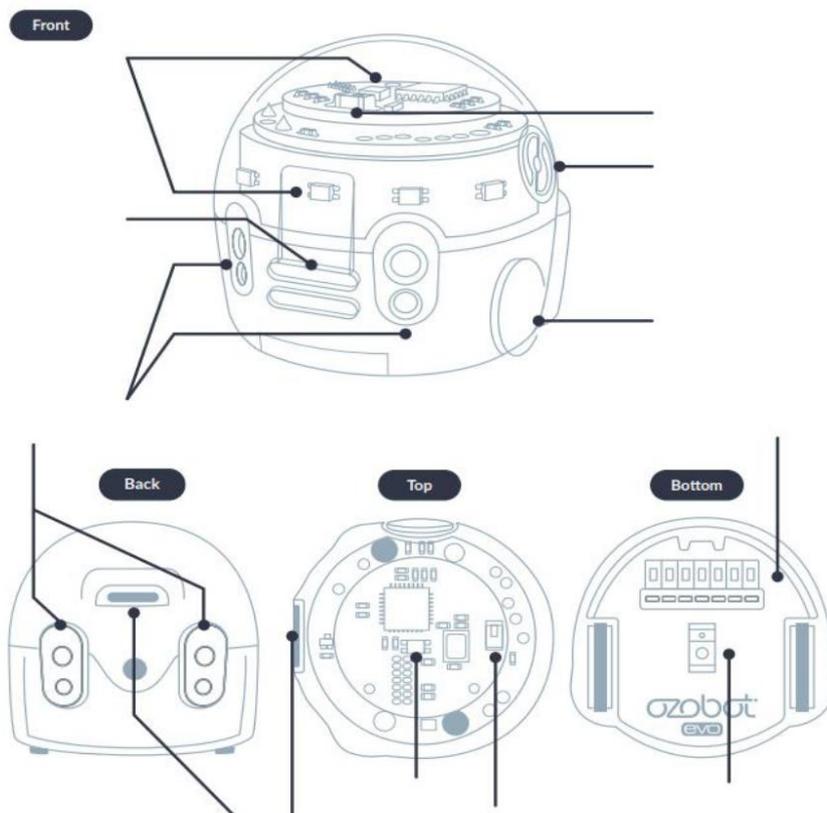
1. Vídeo 1 (1 m) - Introdução
Link: <https://watch.cloudflarestream.com/e5acdf6511b640e8e38d810ca8e17e07>
2. Vídeo 2 (42 s) - O que é o robô Evo?
Link: <https://watch.cloudflarestream.com/0dc28f00e7c4e2fe2428242eadfb7251>
3. Vídeo 3 (1 m) – Componentes de *Hardware* - Sensores
Link: <https://watch.cloudflarestream.com/681a5f0accd2a2b92345ccadd7e6ab91>
4. Vídeo 4 (30 s) – Componentes de *Hardware* - luzes LED
Link: <https://watch.cloudflarestream.com/38b9f2dad88e4cbc6327da391636d3b6>
5. Vídeo 5 (42 s) – Componentes de *Hardware* - Motores, bateria, rodas e microfone
Link: <https://watch.cloudflarestream.com/a9d88cf647c623b3e56b27dd69f1b836>
6. Vídeo 6 (36 s) – Componentes de *Hardware* - Placa mãe e Antena
Link: <https://watch.cloudflarestream.com/8592971980f31ebd2729f70d3ec3e850>
7. Vídeo 7 (35 s) – Componentes de *Hardware* - Porta para carregamento
Link: <https://watch.cloudflarestream.com/d6b456e794d74608cff8179ef765d84c>

8. Vídeo 8 (35 s) – Componentes de *Hardware* - Botão para ligar/desligar o Ozobot
Link: <https://watch.cloudflarestream.com/c36ce72a5a1d23c865191034f2db41f4>

Se estiveste atento, vai ser fácil executar a tarefa seguinte, em que deves preencher com os termos em inglês (*Key Terms*), junto das linhas correspondentes.



- **Key Terms**
- Bluetooth Antenna
 - Color Sensor
 - Line Following Sensors
 - LED Light
 - Proximity Sensors
 - Power Button
 - Speaker
 - Charging Port
 - Wheels/Motor



Segunda parte - Programar o robô Ozobot com cores

Para além da explicação da professora, tens alguns vídeos curtos, sobre como calibrar o robô ozobot e programá-lo através do desenho de cores, sempre que não te lembrares podes recorrer a esta preciosa ajuda!

Vais necessitar também de dois ficheiros para acompanhar a explicação dos vídeos [anexo A](#) e [anexo B](#), solicita a tua professora os documentos de trabalho.

1. Vídeo 1 (43 s) - Introdução
Link: <https://watch.cloudflarestream.com/e5960c834601b80a5ba122ee4fde73fb>

2. Vídeo 2 (1,08 m) - Função do botão ligar do ozobot
Link: <https://watch.cloudflarestream.com/2dc8d090aa9145193881eb89d8bed293>
3. Vídeo 3 (1,03 m) - Como calibrar o ozobot
Link: <https://watch.cloudflarestream.com/e9f39b53a60547f4d036d841df181886>
4. Vídeo 1 (1,33 m) - Função seguir linha – em frente, cantos e curvas
Link: <https://watch.cloudflarestream.com/411167668b418c1fc9bf88aa290ca2bc>
5. Vídeo 1 (1,24 m) - Função seguir linha – Cores
Link: <https://watch.cloudflarestream.com/6e75e34fdf4513d26de764e959b8913a>
6. Vídeo 1 (1,24 m) – Código de cores
Link: <https://watch.cloudflarestream.com/609dd99d7d4650ef0f22f4d74e58f806>
7. Vídeo 1 (1,58 m) - Symmetric Color Codes
Link: <https://watch.cloudflarestream.com/cf8409b4346cb79e0dcccc4f3a522567>
8. Vídeo 1 (2,18 m) - Asymmetric Color Codes
Link: <https://watch.cloudflarestream.com/f53e6a58066238488340d574bb900dcb>



Anexo A

Anexo B

Se ainda tiveres tempo concretiza o percurso do **anexo C**, onde treinas a velocidade do robô ozobot.

😊 Bom trabalho!

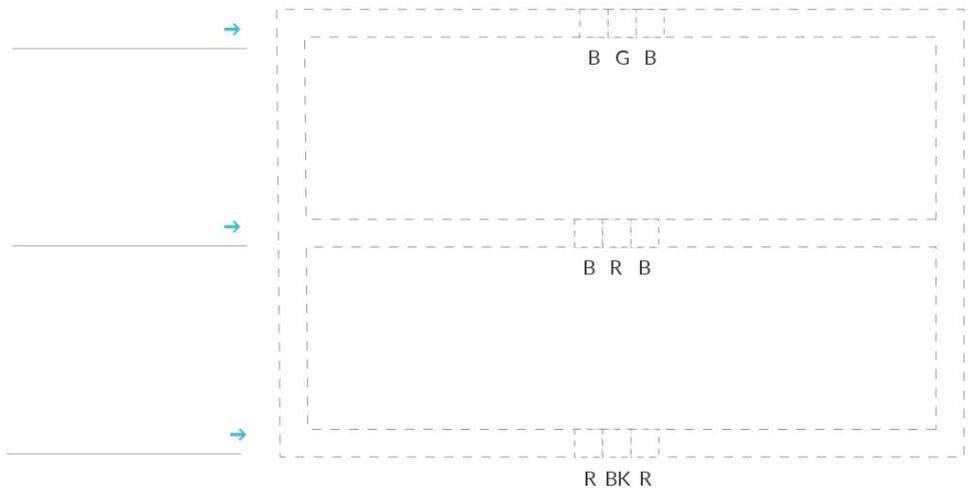
Introduction to Color Codes 01: Basic Training

Name: _____
Date: _____

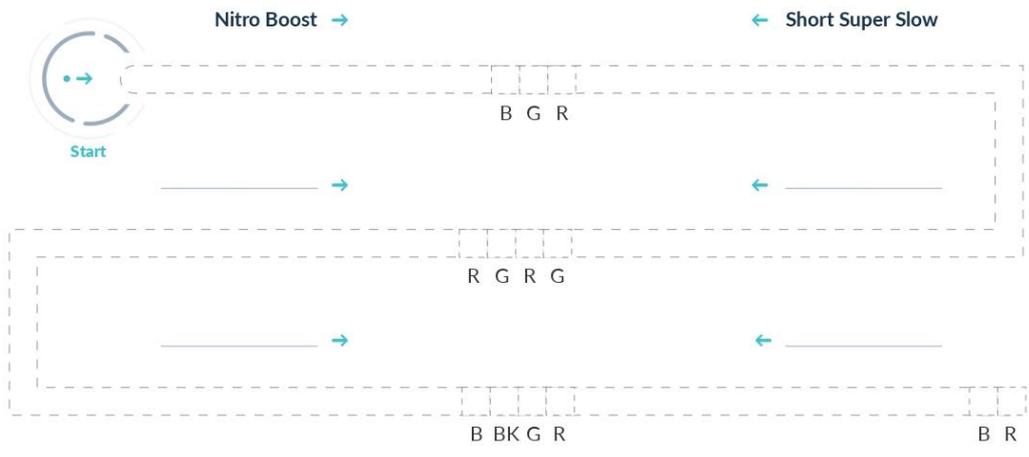
2

Symmetric Codes

The Color Code programs my bot to:



Asymmetric Codes



• Color Key

BK =	B =	R =	G =

Anexo E10 – Programa de intervenção – sessão 10 – Introdução ao Ozobot Blockly

 <p>Agrupamento de Escolas FRANCISCO DE HOLANDA</p> <p>CIÊNCIAS DA COMPUTAÇÃO</p>	ATIVIDADE 10
	OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR) 06/2022 - 50 min. Ano Letivo 2021/2022

Objetivo:
Programar o robô Ozobot através do editor OzoBlockly

Material necessário:

- Ligação à internet
- Robô Ozobot
- Plataforma OzoBlockly



Hoje vais programar o robô Ozobot, de forma alternativa com blocos de programação. Na primeira parte da aula vais aprender a programar determinadas ações e sequências, reconhecendo os blocos disponíveis. Na segunda parte vais explorar exercícios com ciclos (*loop*) e condições.

Primeira parte - Sequências

Deves executar as ações que surgem no vídeo, e estar atento à explicação da professora. Sempre que tiveres dúvidas solicita esclarecimento à professora.

1. Vídeo 1 (12,20 m) - Sequências
Link: <https://watch.cloudflarestream.com/c0229b74fea1b545323ab9ef2ebaa2fa>

2. Acede à plataforma OzoBlockly <https://ozoblockly.com>

3. Limpar o programa existente, clicando no icon 

4. Altera o nome do programa para **sequencias**, clica em  e na opção 

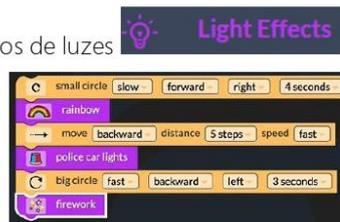
5. Adiciona os seguintes blocos da categoria movimento 



6. Verifica se tens o robô ligado ao tablet/computador e envia o programa para o Ozobot. Observa o seu comportamento, altera os parâmetros dos três blocos anteriores e envia novamente o programa para o robô.

7. A seguir vais inserir blocos da categoria efeitos de luzes

Explora os 8 tipos de efeitos existentes e insere os três blocos apresentados na figura e observa o seu funcionamento enviando o programa para o robô .



8. A seguir vais explorar a categoria de som,

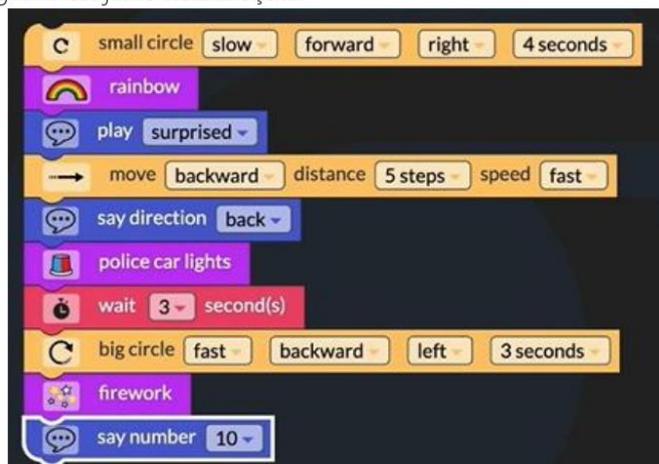


em que podes explorar 5 tipos de sons disponíveis.

9. Altera os parâmetros de três blocos a tua escolha e insere esses sons no código e observa o funcionamento.

10. Por vezes é necessário introduzir um bloco de tempo para fazer um compasso de espera. Para isso usamos o bloco de tempo . Completa o teu código com o seguinte conjunto de instruções:

11. Altera o código com as instruções da figura e envia para o robô. Como podes observar o bloco espera 3 segundos (*wait*) permite introduzir uma pausa.



Segunda parte – Ciclos (*loop*) e condições

Nesta parte da aula, vais relembrar os conceitos sobre ciclos, mas agora com o robô Ozobot. Deves executar as ações que surgem no vídeo, e estar atento à explicação da professora. Sempre que tiveres dúvidas solicita esclarecimento à tua professora.

1. Vídeo 1 (12,20 m) - Ciclos

Link: <https://watch.cloudflarestream.com/59d47ebef29524bd8dd809a41bd57d3a>

2. Acede à plataforma OzoBlockly <https://ozoblockly.com>



3. Limpar o programa existente, clicando no icon



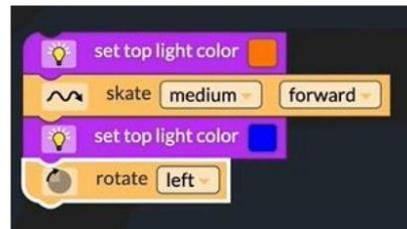
4. Altera o nome do programa para **ciclos**, clica em



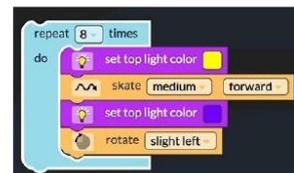
e na opção

5. Constrói a seguinte sequência de blocos, apresentada na imagem do lado direito.

6. Agora vais repetir esta sequência de blocos de forma a desenhar um quadrado. Inserir o bloco que executa 4x esta sequência.



7. Altera a sequência para o robô desenhar um hexágono

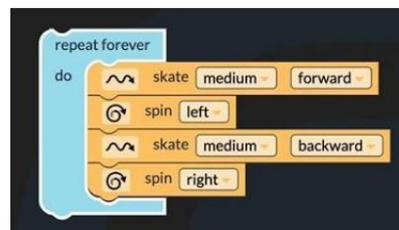


8. Agora vais usar o ciclo de repetição para sempre, ou seja, vai sempre executar uma determinada sequência até alguém parar.

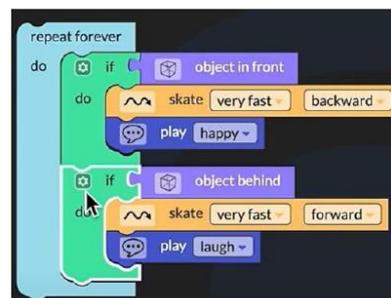
9. No próximo exercício vais criar um exercício com condições, para isso deves escolher o nível três



que dispõem de mais blocos de código.



10. Cria um programa com o nome de **condições**, insere os seguintes blocos e observa como se comporta o robô Ozobot. Como podes constatar o Ozobot usa os sensores de proximidade da frente e de trás. Sempre que ele encontra um objeto à frente anda para trás e emite um som de felicidade, caso contrário se encontrar um objeto nos sensores de trás, anda para a frente e emite um som diferente!



11. Altera o código para que o robô Ozobot siga a tua mão, ou seja sempre que ele encontra um objeto à sua frente (mão) segue-a. Altera o teu código e experimenta essas alterações.

12. Se já conseguiste estás de parabéns, podes agora explorar outros blocos e construir o teu próprio código!

😊 Bom trabalho!

Anexo F – Painel de sessão Recursos Educativos Digitais

Plano de Ação para a Transição Digital	Organização e Colaboração	PTD—Sessões de Curta Duração 5 / 8
<p>o Plano de Ação para a Transição Digital, de 21 de abril de 2020 (Resolução do Conselho de Ministros n.º 30/2020), tem em vista, entre outras, colmatar os défices já diagnosticados nas escolas ao:</p> <ol style="list-style-type: none"> disponibilizar equipamento individual a alunos e professores; garantir a conectividade móvel gratuita para alunos e professores; permitir acesso a recursos educativos digitais de qualidade (p. ex. manuais digitais; repositórios de RED); apostar num plano de capacitação digital de docentes. <p>É essencial que esta última aposta - Capacitação Digital Docente – seja ganha. E tanto mais o será quanta maior for a capacidade de mobilizar os professores na transformação da escola, através de ações de formação /informação assertiva e focalizada nas suas dificuldades, anseios e dúvidas. Este é o propósito do CFFH – aproximar e informar – oferecendo à sua comunidade várias leituras/focos sobre o digital, problematizando escolhas, analisando estratégias, partilhando recursos e projetos, debatendo “lugares-comuns” do discurso tecnológico.</p>	<p>Entidade Organizadora:</p>  <p>Entidades Colaboradoras:</p>  <p>AE Participantes:</p> 	

Objetivos Gerais da Ação	Recursos Educativos Digitais (RED)	Organização
<p><i>A Capacitação Digital das Escolas pretende impulsionar o desenvolvimento de estratégias inovadoras e indutoras de mudança que terão impacto na melhoria da gestão escolar, na qualidade do ensino, nas aprendizagens e nos resultados escolares. (DGE)</i></p> <ul style="list-style-type: none"> Instituir práticas colaborativas facilitadoras da aprendizagem; Analisar recursos educativos digitais de suporte ao PADDE; Desmistificar conceitos para transformar práticas; Promover comunidades de prática, para estimular a reflexão e a colaboração em contexto educativo; Refletir sobre processos organizativos, pedagógicos e tecnológicos que, com recurso a ferramentas e a ambientes digitais, promovam a qualidade do processo educativo. 	<p>Com vista a cumprir os desígnios da DGE, o CFFH propõe uma sessão formativa/informativa, com o objetivo geral de promover Recursos Educativos Digitais de qualidade, de acesso livre, validados científica e pedagogicamente, e de apoiar o seu desenvolvimento nas escolas da sua área geográfica.</p> <p>A utilização destes recursos contribuirá para:</p> <ul style="list-style-type: none"> A inovação e a diferenciação pedagógica; A adoção de estratégias promotoras do desenvolvimento crítico, criativo e autónomo dos alunos; O incentivo a modos de avaliação que promovam a aprendizagem (avaliação pedagógica); A indução de práticas de sala de aula que melhor atendam à prossecução do perfil dos alunos à saída da escolaridade obrigatória. 	<p>Conjunto de duas sessões de 1h30m cada, para constituir uma ação de curta duração de 3h, com requisitos para ser reconhecida, nos termos do Despacho n.º 5741/2015, de 29 de maio:</p> <p>05 de julho—18h – 19h30 Apresentação dos Recursos Educativos Digitais disponíveis na plataforma da DGE.</p> <p>19 de julho—18h – 19h30 Apresentação de práticas com utilização de RED:</p> <ul style="list-style-type: none"> AE Pevidém — Professoras Maria da Conceição Bessa e Cristina Almeida (Projetos dinamizados no 1º CEB) AE Fafe— Renato Silva / Aprendizagem invertida com grupo de especialistas: um exemplo (Projetos dinamizados no 2º CEB) AE Francisco de Holanda— Professores José Carlos Silva, Sandra Machado e Carla Barbosa (Projetos dinamizados no 1º CEB) AEPAS—Professor Luís Silva (Projetos dinamizados no 1º CEB) <p>Destinatários Professores das escolas associadas ao CFFH.</p>

	ENTREVISTA POR FOCUS GROUP OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR)
	Junho - Ano Letivo 2021/2022

PROJETO DE INVESTIGAÇÃO

“Robótica Educativa um contributo para o desenvolvimento do pensamento computacional na escolaridade inicial.”

Este projeto de investigação está a ser desenvolvido no âmbito do mestrado em Ciências da Educação – Área de Especialização em Tecnologia Educativa, da Universidade do Minho (UM) e, tem como objetivo principal caracterizar o processo de desenvolvimento de atividades de Robótica Educativa com crianças do 1.º ciclo do ensino básico, que promovam dimensões do Pensamento Computacional.

Toda a informação recolhida será tratada de forma anónima e confidencial e, só a equipa de investigação terá acesso aos dados.

Por favor, responda sinceramente às questões apresentadas. Obrigado!

ENTREVISTA POR *FOCUS GROUP*

PROFESSORES REponsáveis PELA LECIONAÇÃO DA OFERTA COMPLEMENTAR DE PROGRAMAÇÃO E ROBÓTICA (OCPR) E PROFESSORES TITULARES DE TURMA

1. Na sua opinião as escolas do 1.º ciclo pertencentes ao agrupamento estão equipadas com materiais para o ensino da Robótica Educativa (RE)?
2. A programação e robótica é uma constante neste agrupamento desde 2016, como descreve as diferentes metodologias a implementar. Na sua opinião qual ou quais aquelas que resultam efetivamente?
3. Qual a importância da aprendizagem da Programação e do uso de robôs (Robótica) no 1.º Ciclo?

4. Quais os aspetos mais relevantes que foi observando nas aulas de OCPR da reação dos alunos?
5. Na sua ótica acha que a Robótica Educativa foi considerada como um artefacto transversal em articulação com as restantes áreas curriculares e não como mais uma área curricular?
6. Entres os diferentes projetos que envolveram a robótica, qual ou quais foram os que refletiram na aprendizagem dos alunos nas diferentes disciplinas?
7. As atividades desenvolvidas com recurso à robótica educativa permitiram contribuir para desenvolvimento dos princípios do pensamento computacional?
8. Na sua opinião, enumere dois pontos fortes e dois pontos fracos da implementação da robótica educativa como uma metodologia ativa nas aulas de OCPR?
9. Deverá este projeto da OCPR ter continuidade? Se sim/não porquê?
10. Qual a avaliação que faz da OCPR neste ano letivo?

PROFESSORES RESPONSÁVEIS PELA LECIONAÇÃO DA OFERTA COMPLEMENTAR DE
PROGRAMAÇÃO E ROBÓTICA (OCPR)

1. As atividades desenvolvidas com recurso à robótica educativa permitiram que os alunos aprendessem conceitos das ciências de computação (programar).
2. Dos diferentes kits de robótica educativa trabalhados em sala de aula (Lego Wedo 2.0, Placa Micro:Bit , Robô Ozobot) entre outras aplicações e artefactos, qual foi o grau de exploração?
3. Em que medida as atividades com Robótica Educativa, contribuíram para o desenvolvimento dos alunos ao nível do pensamento do pensamento computacional, mais concretamente nas dimensões (Decomposição, Abstração, Reconhecimento de Padrões, Algoritmos e Depuração)?

Obrigado pela sua colaboração!

Anexo H - Teste de Pensamento Computacional aplicado em formato digital na plataforma Quizizz

INSTRUÇÕES

O teste é composto por 28 perguntas, todas as perguntas têm 4 opções de resposta (A, B, C ou D), das quais apenas uma é a correta.

A partir do momento que comeces o teste, tens 45 minutos para fazeres o melhor que conseguires.

Não é necessário que respondas a todas as perguntas.

Antes de começares o teste, vamos ver 3 exemplos para que te familiarizes com o tipo de perguntas que irás encontrar, nas quais vão aparecer as personagens que agora te apresentamos.



Pac-Man



Fantasma



Artista

Segue um exemplo de uma questão de familiarização, na plataforma on-line (quizz) com a explicitação da resposta correta:

Os alunos têm a possibilidade de fazer Zoom nas imagens, de forma a conseguir obter melhor visibilidade.

Links do Teste

<https://quizizz.com/admin/quiz/6234498af51247001d082cec>

Link das Questões de Treino

<https://quizizz.com/admin/quiz/62113f998765360020177846>

Exemplo 1

Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?

→ → ↓

→ → ↑

→ ↑ ↑

→ ↓ ↓

Exemplo 2

Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?

- mover para a frente
virar à esquerda ◡ ▾
mover para a frente
mover para a frente
- mover para a frente
virar à direita ◡ ▾
mover para a frente
mover para a frente
- mover para a frente
mover para a frente
virar à direita ◡ ▾
mover para a frente
- mover para a frente ✓
mover para a frente
virar à esquerda ◡ ▾
mover para a frente

Exemplo 3

Quais os passos que o Artista deve executar para desenhar a figura? O lado menor mede 50 pixéis e o maior mede 100 pixéis.

- mover para a frente por 50 pixéis
virar à esquerda por 90 graus
mover para a frente por 100 pixéis
- mover para a frente por 50 pixéis
virar à direita por 90 graus
mover para a frente por 100 pixéis
- mover para a frente por 100 pixéis
virar à esquerda por 90 graus
mover para a frente por 50 pixéis
- mover para a frente por 100 pixéis
virar à direita por 90 graus
mover para a frente por 50 pixéis

Questão 1

Quais os passos que levam o Pac-Man até ao Fantasma pelo caminho assinalado?

- → ↓
- ↓ ↓
- → ↓ ↓
- ↓ ↓ →

Questão 5



Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

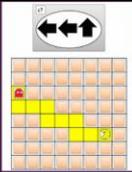
× 5

× 3

× 4

× 2

Questão 6



Quantas vezes se deve repetir a seguinte sequência para levar o Pac-Man até ao Fantasma pelo caminho assinalado?

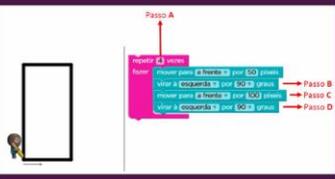
X 2

X 1

X 4

X 3

Questão 7



Para que o Artista desenhe uma vez o seguinte retângulo (50 pixéis de largura e 100 pixéis de altura), qual o passo da sequência que está incorreto?

PASSO A

PASSO B

PASSO C

PASSO D

Questão 8

Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

repetir 4 vezes

fazer

repetir 3 vezes

fazer

mover para a frente

virar à direita

mover para a frente

repetir 3 vezes

fazer

repetir 4 vezes

fazer

mover para a frente

virar à direita

mover para a frente

repetir 3 vezes

fazer

repetir 4 vezes

fazer

mover para a frente

virar à direita

mover para a frente

repetir 4 vezes

fazer

mover para a frente

repetir 4 vezes

fazer

virar à direita

mover para a frente

Questão 9

Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

Repetir até chegar ao

Repetir até chegar ao

Repetir até chegar ao

Repetir até chegar ao

Questão 10

Qual o passo que falta na seguinte sequência para que o Pac-Man vá até ao Fantasma pelo caminho assinalado?

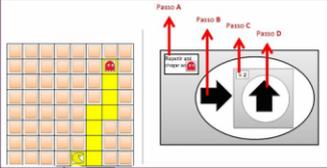
virar à esquerda

virar à direita

mover para a frente

Não falta nenhum passo

Questão 11



Para que o Pac-Man chegue até ao Fantasma pelo caminho assinalado, qual o passo da seguinte sequência que está incorreto?

PASSO A PASSO B PASSO C PASSO D

Questão 12



Qual a sequência que faz com que o Artista desenhe a escada até à flor? Cada degrau sobe 30 pixéis.

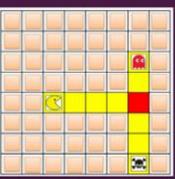
repetir até à flor
fazer
repetir 3 vezes
fazer
mover para a frente 30 por 30 pixels
virar à direita 90 graus
saltar para a frente 30 por 30 pixels

repetir até à flor
fazer
repetir 4 vezes
fazer
mover para a frente 30 por 120 pixels
virar à direita 90 graus
saltar para a frente 30 por 30 pixels

repetir até à flor
fazer
repetir 4 vezes
fazer
mover para a frente 30 por 30 pixels
virar à direita 90 graus
saltar para a frente 30 por 210 pixels

repetir até à flor
fazer
repetir 7 vezes
fazer
mover para a frente 30 por 30 pixels
virar à direita 90 graus
saltar para a frente 30 por 30 pixels

Questão 13



Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

Repetir até chegar ao 
se passar por  

Repetir até chegar ao 
se passar por   

Repetir até chegar ao 
se passar por  

Repetir até chegar ao 
se passar por   

Questão 14

Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

repetir até

fazer mover para a frente

se há caminho para a direita

fazer virar à direita

repetir até

fazer virar à direita

se há caminho para a direita

fazer mover para a frente

repetir até

fazer mover para a frente

se há caminho para a direita

fazer virar à esquerda

repetir até

fazer mover para a frente

se há caminho para a esquerda

fazer virar à esquerda

Questão 15

O que falta na seguinte sequência para que o Pac-Man vá até ao Fantasma pelo caminho assinalado?

Opção A

Opção B

Opção C

Tanto a opção A como a opção C estão corretas

Questão 16

Para que o Pac-Man chegue até ao Fantasma pelo caminho assinalado, qual o passo da seguinte sequência que está incorreto?

PASSO A

PASSO B

PASSO C

PASSO D

Questão 17

Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

repetir até

fazer se há caminho em frente

fazer mover para a frente

se não virar à esquerda

repetir até

fazer se há caminho em frente

fazer mover para a frente

se não virar à direita

repetir até

fazer se há caminho para a direita

fazer virar à direita

se não mover para a frente

repetir até

fazer se há caminho para a esquerda

fazer virar à esquerda

se não mover para a frente

Questão 18

Qual a sequência que leva o Pac-Man até ao Fantasma pelo caminho assinalado?

repetir até

fazer se há caminho em frente

fazer mover para a frente

se não virar à esquerda

repetir até

fazer se há caminho em frente

fazer mover para a frente

se não virar à direita

repetir até

fazer se há caminho para a direita

fazer virar à direita

se não mover para a frente

repetir até

fazer se há caminho para a esquerda

fazer virar à esquerda

se não mover para a frente

Questão 19

repetir até

fazer se há caminho em frente → Passo A

fazer mover para a frente → Passo B

se não se há caminho para a direita → Passo C

fazer virar à esquerda → Passo D

se não virar à direita

Para que o Pac-Man chegue até ao Fantasma pelo caminho assinalado, qual o passo da seguinte sequência que está incorreto?

PASSO A

PASSO B

PASSO C

PASSO D

Questão 20



Qual o passo que falta na seguinte sequência para que o Pac-Man vá até ao Fantasma pelo caminho assinalado?

mover para a frente

virar à direita ↻

virar à esquerda ↻

Não falta nenhum passo

Questão 21



Qual a sequência que leva o Pac-Man até aos Morangos pelo caminho assinalado e faz com que ele coma o número de Morangos indicado?

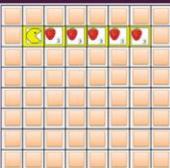
enquanto houver caminho em frente
fazer mover para a frente
repetir 3 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer mover para a frente
repetir 4 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer mover para a frente
repetir 5 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer mover para a frente
repetir 3 vezes
fazer comer 1 morango

Questão 22



Qual a sequência que leva o Pac-Man até aos Morangos pelo caminho assinalado e faz com que ele coma o número de Morangos indicado?

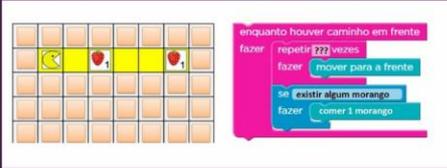
enquanto houver caminho em frente
fazer repetir 5 vezes
fazer mover para a frente
repetir 3 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer mover para a frente
repetir 3 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer repetir 3 vezes
fazer mover para a frente
repetir 5 vezes
fazer comer 1 morango

enquanto houver caminho em frente
fazer mover para a frente
repetir 3 vezes
fazer comer 1 morango

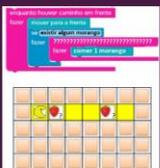
Questão 23



O que falta na seguinte sequência para que o Pac-Man vá pelo caminho assinalado comendo o número de Morangos indicado?

1 VEZ 2 VEZES 3 VEZES 5 VEZES

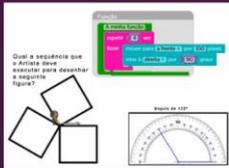
Questão 24



O que falta na seguinte sequência para que o Pac-Man vá pelo caminho assinalado comendo o número de Morangos indicado (número desconhecido)?

enquanto houver caminho em frente enquanto não houver caminho em frente enquanto houver morangos enquanto não houver morangos

Questão 25



Se tivermos o seguinte conjunto de passos a que chamamos de "A minha função", e que desenha um quadrado com 100 pixéis de lado:

repetir 3 vez, fazer A minha função, virar à direita por 120 graus

repetir 3 vez, fazer A minha função, virar à direita por 120 graus

repetir 4 vez, fazer A minha função, virar à direita por 90 graus

repetir 4 vez, fazer A minha função, virar à direita por 90 graus

Questão 26



O que falta na seguinte sequência para que o Artista desenhe a seguinte figura?

Se tivermos o seguinte conjunto de passos a que chamamos de “A minha função” e que desenha um triângulo com 50 pixéis de lado:

15

5

4

3

Questão 27



Qual a sequência que leva o Pac-Man em um Morango para o próximo Morango comê-lo? Qual o número de Morangos correspondente?

Se tivermos o seguinte conjunto de passos a que chamamos de “comer 5”:

```

mover para a frente
virar à direita
repetir 3 vezes
  fazer
    mover para a frente
    comer 5

```

```

mover para a frente
virar à direita
repetir 3 vezes
  fazer
    comer 5
mover para a frente

```

```

mover para a frente
virar à direita
repetir 5 vezes
  fazer
    mover para a frente
    comer 5

```

```

mover para a frente
virar à direita
repetir 5 vezes
  fazer
    comer 5
mover para a frente

```

Questão 28



O que falta na seguinte sequência para que o Pac-Man vá pelo caminho indicado comendo todos os Morangos?

Se tivermos o seguinte conjunto de passos a que chamamos de “andar e comer 4”:

3

4

5

6

