



PLANO CURRICULAR

Nível II

“Programação de Histórias”



A metodologia ubbu

A ubbu é uma solução para o ensino de ciência da computação, programação e desenvolvimento do pensamento computacional. Pensada para capacitar qualquer professor a aprender e ensinar sobre estas matérias, a plataforma faz uso das STEAM (*Science, Technology, Engineering, Arts e Mathematics*), para interligar conhecimentos e metodologias, e dos [Objetivos de Desenvolvimento Sustentável](#) das Nações Unidas, para consciencializar o aluno quanto ao seu papel no mundo.

Elementos da metodologia ubbu



Ciência da computação

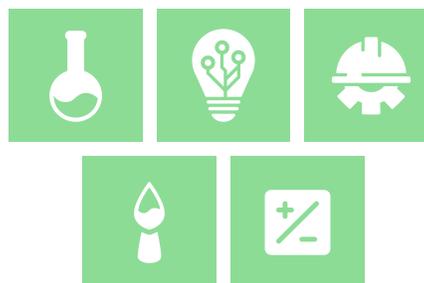
A ciência da computação é a área que estuda os computadores e a forma como os utilizamos para manipular informação. O seu impacto faz-se sentir em todas as áreas da nossa sociedade, facilitando o acesso à informação, acelerando processos, interligando pessoas e negócios, etc. Compreender as fundamentações da ciência da computação, e as suas aplicações, torna-se cada vez mais essencial para compreender o mundo e uma sociedade cada vez mais digital.

Programação

A programação faz parte da ciência da computação e é uma das formas de desenvolver o pensamento computacional, que trabalha diversas competências transversais. A sua natureza prática acelera a aprendizagem e abre novas possibilidades de invenção e de resolução de problemas, sendo uma ferramenta essencial de um aluno preparado para o futuro.

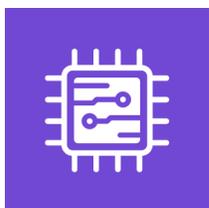
STEAM

O uso das STEAM faz-se a dois níveis: na interdisciplinaridade, relacionando a matéria com ciência, matemática, artes, engenharia ou tecnologia e nos projetos, onde metodologia é aplicada a momentos de desafio aberto em que é apresentado um problema aos alunos e onde têm liberdade para interpretar e resolver a questão, recorrendo a todas as ferramentas e aprendizagens obtidas.



ODS

Os Objetivos de Desenvolvimento Sustentável são 17 objetivos adotados por todos os Estados-Membros das Nações Unidas para 2030. Pretendem transmitir um apelo urgente à ação de todas as nações em criar um mundo sustentável. Ao introduzir estes temas na aprendizagem, evidenciamos o papel que cada aluno tem na criação de um planeta mais justo, saudável e sustentável.



Competências

O ensino de ciência da computação e programação desenvolve nos alunos várias competências de pensamento computacional essenciais, como o reconhecimento de padrões ou algoritmia. A acrescentar-se a estas o leque de atividades do currículo também desenvolve competências psicomotoras, essenciais para o futuro do aluno, como o pensamento crítico ou a colaboração. Estas competências foram desenvolvidas tendo como base o *Perfil dos Alunos para o Século XXI* (2017).

Progressão curricular ubbu

Visão geral do progresso dos alunos nos três primeiros níveis de currículo



Nível I

“Introdução à ciência da computação”

Apresentação dos conceitos básicos de ciência da computação e programação.

Nível II

“Programação de histórias”

Ensino da programação por blocos através da criação de histórias e da exploração de conceitos de ciência da computação.

Nível II

“Programação de jogos”

Ensino da programação de blocos mais complexos através da criação de jogos e aprofundamento de conceitos de ciência da computação.



Plano Curricular

Neste documento, encontram-se sintetizadas as aprendizagens esperadas a nível de ciência da computação, programação, as STEAM (*Science, Technology, Engineering, Arts e Mathematics*), bem como as competências (computacionais e psico-motoras) trabalhadas por estas atividades. As aulas expandem-se através dos Objetivos do Desenvolvimento Sustentável, temas que sensibilizam os alunos para a importância de um mundo mais sustentável.

Cada nível de currículo da ubbu é composto por 30 aulas, com duração média de uma hora cada. Este número de aulas pretende cobrir um ano letivo, mas sugere-se ao professor que adapte este conteúdo ao ritmo de aprendizagem da turma.



Currículo Nível II — “Programação de histórias”

Neste currículo é explorado o ensino da programação por blocos através da criação de histórias e da dinamização de conceitos da ciência da computação.

Descrição

O nível II proporciona aos alunos a aprendizagem da programação básica de desenvolvimento de programas e resolução de problemas. Vão aprender utilizando uma ferramenta de programação com blocos, para poderem manipular estes conceitos de maneira prática. Em complemento, vão aprender conteúdos de ciência da computação que exploram alguns conceitos fundamentais e as suas origens.

A ciência, tecnologia e matemática são as áreas de STEAM com grande foco neste nível. Levamos a atenção dos alunos até aos seguintes Objetivos de Desenvolvimento Sustentável: 7 - Energias renováveis e acessíveis; 13 - Ação climática; 14 - Proteção da vida marinha; 15 - Proteção da vida terrestre. Temas incontornáveis numa altura em que os efeitos do aquecimento global se manifestam cada vez mais pelo planeta.

Metodologia

Este nível apresenta uma nova metodologia de aprendizagem. O currículo dos alunos terá uma abordagem modular. Ou seja, está organizado em conjuntos de aulas dedicados a determinados conceitos que, por sua vez, se dividem em três momentos de aprendizagem: o primeiro, onde aprendem o funcionamento de cada bloco de programação; o segundo, onde interligam os blocos de programação entre si para perceber em que contextos podem ser usados; e o terceiro, onde vão responder a desafios, criando projetos livres em que os alunos aplicam as aprendizagens realizadas até ao momento. Estes três momentos de aprendizagem estão presentes em todos os módulos excepto o primeiro, que serve como introdução.

No módulo final, os alunos são desafiados a resolver um último problema aberto, relacionado com os Objetivos de Desenvolvimento Sustentável destacados neste nível. Nesta fase final, todas as competências e aprendizagens são aplicadas para a conclusão do desafio.

Objetivos

No final deste nível, os alunos devem ser capazes de:

1. Utilizar e manipular blocos simples para criar histórias na ferramenta de criação de projetos.
2. Resolver desafios abertos utilizando a programação de blocos.
3. Explicar e exemplificar conhecimentos de ciência da computação e a sua história.



Mapa do Nível II

Visão sintetizadas das aprendizagens esperadas ao longo do nível.

Módulo	Objetivos Pedagógicos	Conceitos de Ciência da Computação	STEAM	Objetivos de Desenvolvimento Sustentável das Nações Unidas	Competências	
					Pensamento Computacional	Cognitivo-motoras
A. Introdução Aula 1	Navegar pela interface da ubbox, percebendo as funções gerais do seu funcionamento.	Programação	Tecnologia	4 - Educação de qualidade		Coordenação olho-mão
B. Movimentos, Eventos e Controlo Aulas 2-5	Explicar o conceito de algoritmo. Explicar o conceito de <i>bugs</i> . Explicar quem foram algumas personalidades e o seu papel na história da computação. Utilizar blocos de eventos. Utilizar blocos de movimentos. Utilizar blocos de controlo..	<i>Bugs</i> Controlo Cultura Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas Modularidade	Ciência Tecnologia Engenharia Matemática	2 - Erradicar a fome 3 - Saúde de qualidade 5 - Igualdade de género 6 - Água potável e saneamento 9 - Indústria, inovação e infraestrutura 11 - Cidades e comunidades sustentáveis 13 - Ação climática	Algoritmos Decomposição Reconhecimento de padrões	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade perceptivo motora Responsabilidade ambiental
C. Estilo Aulas 6-8	Explicar o que é o código binário e transformar informação simples nesta linguagem. Utilizar blocos de estilo.	Código Binário Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas	Tecnologia Engenharia	3 - Saúde de qualidade 5 - Igualdade de género 9 - Indústria, inovação e infraestrutura 11 - Cidades e comunidades sustentáveis 16 - Paz, justiça e instituições eficazes	Algoritmos Decomposição Reconhecimento de padrões	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento



						Compreensão e expressão Capacidade percetivo motora Colaboração
D. Caneta Aulas 9–11	Explicar o que são píxeis e como formam imagens. Explicar o que é RGB e relacionar com píxeis. Utilizar blocos de caneta.	RGB Píxeis Cultura Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas	Tecnologia Engenharia Artes Matemática	7 - Energias renováveis e acessíveis 9 - Indústria, inovação e infraestrutura 11- Cidades e comunidades sustentáveis 12 - Consumo e produção sustentáveis	Algoritmos Decomposição Reconhecimento de padrões	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade percetivo motora Coordenação olho-mão
E. Eventos e Estilo Aulas 12–15	Explicar os conceitos de input e output, enumerando dispositivos como exemplo. Diferenciar dispositivos analógicos de dispositivos digitais. Utilizar blocos de mensagens (Eventos). Utilizar blocos de estilo.	Input e output Analógico e digital Dispositivos Controlo Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas	Ciência Tecnologia Engenharia	3 - Saúde de qualidade 7 - Energias renováveis e acessíveis 15 - Proteção da vida terrestre	Algoritmos Decomposição	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade percetivo motora Colaboração Responsabilidade ambiental
F. Movimentos Aulas 16–19	Utilizar blocos de movimento, relacionados com ângulos.	Cultura Algoritmos Resolução de problemas	Ciência Tecnologia Engenharia	1 - Erradicar a pobreza 6 - Água potável e saneamento 7 - Energias renováveis	Algoritmos Decomposição	Pensamento crítico Resolução de problemas Criatividade



	Identificar graus de ângulos.	Modularidade Desenvolvimento de programas	Artes Matemática	e acessíveis 9 - Indústria, inovação e infraestruturas 10 - Reduzir as desigualdades 11 - Cidades e comunidades sustentáveis 13 - Ação climática		Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade perceptivo motora Responsabilidade social
G. Movimentos, e Estilo Aulas 20–23	Situar coordenadas num eixo cartesiano. Explicar o conceito da aprendizagem automática e as suas vantagens. Utilizar blocos de movimento relacionados com coordenadas. Utilizar blocos de estilo relacionados com coordenadas.	Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas	Ciência Tecnologia Engenharia Matemática	1 - Erradicar a pobreza 7 - Energias renováveis e acessíveis 9 - Indústria, inovação e infraestruturas 10 - Reduzir as desigualdades 11 - Cidade e comunidades sustentáveis 13 - Ação climática 15 - Proteger a vida terrestre	Algoritmos Decomposição	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade perceptivo motora Colaboração Responsabilidade social
H. Controlo, Operadores e Sensor Aulas 24–28	Explicar e utilizar o conceito de condições. Utilizar blocos de condições (Controlo). Utilizar blocos de operadores. Utilizar blocos de sensor.	Condições Cultura Algoritmos Modularidade Resolução de problemas Desenvolvimento de programas Controlo	Ciência Tecnologia Arte Matemática	2 - Erradicar a fome 5 - Igualdade de género 7 - Energias renováveis e acessíveis 9 - Indústria, inovação e infraestruturas 12 - Produção e consumo sustentáveis 13 - Ação climática 14 - Proteção da vida marinha 15 - Proteção da vida terrestre	Algoritmos Decomposição	Pensamento crítico Resolução de problemas Criatividade Participação Transformação de informação em conhecimento Compreensão e expressão Capacidade perceptivo motora Responsabilidade ambiental

<p>I. Projetos Finais</p> <p>Aulas 29-30</p>	<p>Planear, criar e apresentar a resposta a um desafio na ubbox.</p> <p>Utilizar blocos de eventos.</p> <p>Utilizar blocos de movimento.</p> <p>Utilizar blocos de controlo.</p> <p>Utilizar blocos de estilo.</p> <p>Utilizar blocos de caneta.</p> <p>Utilizar blocos de operadores.</p> <p>Utilizar blocos de sensor.</p>	<p>Cultura</p> <p>Algoritmos</p> <p>Resolução de problemas</p> <p>Desenvolvimento de programas</p>	<p>Ciência</p> <p>Tecnologia</p> <p>Engenharia</p>	<p>6 - Água potável e saneamento</p> <p>7 - Energias renováveis e acessíveis</p> <p>9 - Indústria, inovação e infraestruturas</p> <p>11 - Cidades e comunidades sustentáveis</p> <p>13 - Ação climática</p> <p>14 - Proteção da vida marinha</p> <p>15 - Proteção da vida terrestre</p>	<p>Algoritmos</p> <p>Decomposição</p>	<p>Pensamento crítico</p> <p>Resolução de problemas</p> <p>Criatividade</p> <p>Participação</p> <p>Transformação de informação em conhecimento</p> <p>Compreensão e expressão</p> <p>Capacidade percetivo motora</p>
--	--	--	--	---	---------------------------------------	--

Nível II

Módulo A — Introdução

Descrição

Manipular a interface da ubbox, a ferramenta de criação de projetos da ubbu.

Aulas

A1

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Navegar pela interface da ubbox, percebendo as funções gerais do seu funcionamento.

Ciência da computação

- Programação

STEAM

- Tecnologia

Objetivos de Desenvolvimento Sustentável

- 4 - Educação de qualidade

Módulo B — Movimentos, Eventos e Controlo

Descrição

Criar as primeiras animações simples com blocos. Aprender sobre a história da programação, relacionada com algoritmos e *bugs*.

Aulas

A2 a A5

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Explicar o conceito de algoritmo.
2. Explicar o conceito de *bugs*.
3. Explicar quem foram algumas personalidades e o seu papel na história da computação.
4. Utilizar blocos de eventos.
5. Utilizar blocos de movimentos.
6. Utilizar blocos de controlo.

Ciência da computação

- *Bugs*
- Controlo
- Cultura
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Reconhecimento de padrões
- Decomposição

STEAM

- Ciência
- Tecnologia
- Engenharia
- Matemática

Objetivos de Desenvolvimento Sustentável

- 2 - Erradicar a fome
- 3 - Saúde de qualidade
- 5 - Igualdade de género
- 6 - Água potável e saneamento
- 9 - Indústria, inovação e infraestrutura
- 11 - Cidades e comunidades sustentáveis
- 13 - Ação climática

Módulo C — Estilo

Descrição

Personalizar *sprites*, criar diálogos entre *sprites*. Aprender sobre código binário.

Aulas

A6 a A8

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Explicar o que é o código binário e transformar informação simples nesta linguagem.
2. Utilizar blocos de estilo.

Ciência da computação

- Código Binário
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Decomposição

STEAM

- Tecnologia
- Engenharia

Objetivos de Desenvolvimento Sustentável

- 3 - Saúde de qualidade
- 5 - Igualdade de género
- 9 - Indústria, inovação e infraestrutura
- 11- Cidades e comunidades sustentáveis
- 16 - Paz, justiça e instituições eficazes

Módulo D — Caneta

Descrição

Programar *sprites* para criar desenhos. Aprender sobre a resolução e cor no ecrã, através dos píxeis e RGB.

Aulas

A9 a A11

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Explicar o que são píxeis e como formam imagens.
2. Explicar o que é RGB e relacionar com píxeis.
3. Utilizar blocos de caneta.

Ciência da computação

- RGB
- Píxeis
- Cultura
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Reconhecimento de padrões
- Decomposição

STEAM

- Tecnologia
- Engenharia
- Artes
- Matemática

Objetivos de Desenvolvimento Sustentável

- 7 - Energias renováveis e acessíveis
- 9 - Indústria, inovação e infraestrutura
- 11- Cidades e comunidades sustentáveis
- 12 - Consumo e produção sustentáveis

Módulo E — Eventos e Estilo

Descrição

Definir ações de *sprites* com eventos de mensagens. Aprender sobre *input* e *output*. Explorar dispositivos analógicos e digitais.

Aulas

A12 a A15

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Explicar os conceitos de *input* e *output*, enumerando dispositivos como exemplo.
2. Diferenciar dispositivos analógicos de dispositivos digitais.
3. Utilizar blocos de mensagens (Eventos).
4. Utilizar blocos de estilo.

Ciência da computação

- *Input* e *output*
- Analógico e digital
- Dispositivos
- Controlo
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Decomposição

STEAM

- Ciência
- Tecnologia
- Engenharia

Objetivos de Desenvolvimento Sustentável

- 3 - Saúde de qualidade
- 7 - Energias renováveis e acessíveis
- 15 - Proteção da vida terrestre

Módulo F — Movimentos

Descrição

Movimentar *sprites* com ângulos. Conhecer a aprendizagem automática.

Aulas

A16 a A19

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Utilizar blocos de movimento, relacionados com ângulos.
2. Identificar graus de ângulos.

Ciência da computação

- Cultura
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Decomposição

STEAM

- Ciência
- Tecnologia
- Engenharia
- Artes
- Matemática

Objetivos de Desenvolvimento Sustentável

- 1 - Erradicar a pobreza
- 6 - Água potável e saneamento
- 7 - Energias renováveis e acessíveis
- 9 - Indústria, inovação e infraestruturas
- 10 - Reduzir as desigualdades
- 11 - Cidades e comunidades sustentáveis
- 13 - Ação climática

Módulo G — Movimento e Estilo

Descrição

Definir posições de *sprites* e caixas de texto com coordenadas.

Aulas

A20 a A23

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Situar coordenadas num eixo cartesiano.
2. Explicar o conceito da aprendizagem automática e as suas vantagens.
3. Utilizar blocos de movimento relacionados com coordenadas.
4. Utilizar blocos de estilo relacionados com coordenadas.

Ciência da computação

- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Decomposição

STEAM

- Ciência
- Tecnologia
- Engenharia
- Matemática

Objetivos de Desenvolvimento Sustentável

- 1 - Erradicar a pobreza
- 7 - Energias renováveis e acessíveis
- 9 - Indústria, inovação e infraestruturas
- 10 - Reduzir as desigualdades
- 11 - Cidade e comunidades sustentáveis
- 13 - Ação climática
- 15 - Proteger a vida terrestre

Módulo H — Controlo, Operadores e Sensor

Descrição

Identificar e criar condições. Criar animações com condições para levar um algoritmo a tomar diferentes direções.

Aulas

A24 a A28

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Explicar e utilizar o conceito de condições.
2. Utilizar blocos de condições (Controlo).
3. Utilizar blocos de operadores.
4. Utilizar blocos de sensor.

Ciência da computação

- Condições
- Controlo
- Cultura
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Modularidade
- Decomposição

STEAM

- Ciência
- Tecnologia
- Arte
- Matemática

Objetivos de Desenvolvimento Sustentável

- 2 - Erradicar a fome
- 5 - Igualdade de género
- 7 - Energias renováveis e acessíveis
- 9 - Indústria, inovação e infraestruturas
- 12 - Produção e consumo sustentáveis
- 13 - Ação climática
- 14 - Proteção da vida marinha
- 15 - Proteção da vida terrestre

Módulo I — Projetos Finais

Descrição

Responder a desafios de resposta aberta aplicando todas as aprendizagens realizadas ao longo do nível.

Aulas

A29 a A30

Objetivos

No final deste módulo, os alunos devem ser capazes de:

1. Planear, criar e apresentar a resposta a um desafio na ubbox.
2. Utilizar blocos de eventos.
3. Utilizar blocos de movimento.
4. Utilizar blocos de controlo.
5. Utilizar blocos de estilo.
6. Utilizar blocos de caneta.
7. Utilizar blocos de operadores.
8. Utilizar blocos de sensor.

Ciência da computação

- Cultura
- Algoritmos
- Resolução de problemas
- Desenvolvimento de programas
- Decomposição

STEAM

- Ciência
- Tecnologia
- Engenharia

Objetivos de Desenvolvimento Sustentável

- 6 - Água potável e saneamento
- 7 - Energias renováveis e acessíveis
- 9 - Indústria, inovação e infraestruturas
- 11 - Cidades e comunidades sustentáveis
- 13 - Ação climática
- 14 - Proteção da vida marinha
- 15 - Proteção da vida terrestre

Variantes do Nível II

O currículo de nível II da ubbu foi planeado para que alunos entre o 2.º e o 6.º anos de escolaridade possam explorar os mesmos conteúdos de programação e ciência da computação. Existem três variantes do currículo de nível II e cada uma destas proporciona aprendizagens adaptadas às diferentes fases de desenvolvimentos dos alunos.

Estas adaptações correspondem à complexidade e quantidade de conteúdos, ao modo como são apresentados e aos temas abordados a nível de currículo escolar: Quando apresentamos conceitos novos, simplificamos a leitura e complexidade dos textos para os mais novos; os exercícios têm temas e desafios adaptados às aprendizagens escolares e os quizzes de escolha múltipla variam no número de hipóteses e complexidade destas.



Mapa das variantes do Nível II

O mapa das variantes marca as diferenças que existem entre cada versão do currículo, sublinhando apenas os materiais que muda de uma versão para outra e em que aspectos.

Módulo	Aula	2.º Ano	3.º e 4.º Ano	5.º e 6.º Ano
A. Introdução	1	Sem variantes.	Sem variantes.	Sem variantes.
	2	Jogo Carro Robô: 2 níveis. Quiz: 6 perguntas com 3 hipóteses. Galeria "Ada Lovelace": Leitura e informação simplificada.	Jogo Carro Robô: 2 níveis. Quiz: 10 perguntas com 4 hipóteses	Jogo Carro Robô: 4 níveis. Quiz: 10 perguntas com 4 hipóteses.
	3	Quiz: 8 perguntas com 3 hipóteses. Exercício ubbox 4: Desafio simplificado. Galeria "Grace Hopper": leitura e informação simplificada.	Quiz: 10 perguntas com 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses (2 perguntas avançadas).
	4	Jogo Robô SOS: 3 níveis. Exercício ubbox 4: Programar o percurso de um robô ajudante. Quiz: 8 perguntas com 3 hipóteses.	Jogo Foguetões: 5 níveis. Exercício ubbox 4: Programar a viagem Apollo 11 com uma repetição simples. Quiz: 10 perguntas com 4 hipóteses.	Jogo Foguetões: 5 níveis. Exercício ubbox 4: Programar a viagem Apollo 11 e um satélite a mover-se no espaço com repetições complexas. Quiz: 10 perguntas com 4 hipóteses.
	5	Projeto ubbox: Projeto sobre um Robô SOS. Quiz: 8 perguntas com 3 hipóteses.	Projeto ubbox: Projeto sobre a primeira viagem do aluno ao espaço. Quiz: 10 perguntas com 4 hipóteses.	Projeto ubbox: Projeto sobre a primeira viagem do aluno ao espaço. Quiz: 10 perguntas com 4 hipóteses (uma pergunta avançada).
C. Estilo	6	Jogo "Binário": 2 níveis Quiz: 8 perguntas com 3 hipóteses.	Jogo "Binário": 3 níveis Quiz: 10 perguntas com 4 hipóteses.	Jogo "Binário": 4 níveis Quiz: 10 perguntas com 4 hipóteses (1 pergunta avançada).
	7	Exercício ubbox 1: Completar o programa do Pascal a ligar um quadro interativo e a fazer-lhe uma pergunta. Exercício ubbox 2: Ajudar a Ruby a fugir programando teclas. Quiz: 6 perguntas com 3 hipóteses.	Exercício ubbox 1: Programar o Pascal a andar até um quadro interativo, a ligá-lo e a fazer-lhe uma pergunta. Exercício ubbox 2: Ajudar a Ruby a fugir programando teclas. Quiz: 7 perguntas com 4 hipóteses.	Exercício ubbox 1: Programar o Pascal a andar até um quadro interativo, a ligá-lo e a fazer-lhe uma pergunta. Exercício ubbox 2: Ajudar a Ruby a fugir programando teclas. Programar outras personagens a falar uma a seguir à outra. Quiz: 8 perguntas com 4 hipóteses (2 perguntas avançadas).

	8	Quiz: 7 perguntas com 3 hipóteses.	Quiz: 8 perguntas com 4 hipóteses.	Quiz: 8 perguntas com 4 hipóteses.
D. Caneta	9	Quiz: 8 perguntas com 3 hipóteses. Jogo "Jogo dos Píxeis": 2 níveis Galeria "Píxeis e RGB": Leitura e informação simplificada.	Quiz: 9 perguntas com 4 hipóteses. Jogo "Jogo dos Píxeis": 4 níveis	Quiz: 10 perguntas com 4 hipóteses (1 pergunta avançada). Jogo "Jogo dos Píxeis": 4 níveis
	10	Quiz: 7 perguntas com 3 hipóteses. Exercício ubbox 1: Desenhar uma parede quadrada.	Quiz: 8 perguntas com 4 hipóteses. Exercício ubbox 1: Desenhar uma parede quadrada.	Quiz: 10 perguntas com 4 hipóteses (1 pergunta avançada). Exercício ubbox 1: Desenhar uma casa, com uma parede retangular e uma porta quadrada.
	11	Quiz: 8 perguntas com 2 e 3 hipóteses.	Quiz: 10 perguntas com 3 e 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses.
E. Eventos e Estilo	12	Quiz: 7 perguntas com 2 e 3 hipóteses. Jogo "Puzzle de Blocos": 2 níveis Imagem "Mensagens": Leitura e informação simplificada.	Quiz: 8 perguntas com 4 hipóteses. Jogo "Puzzle de Blocos": 3 níveis	Quiz: 8 perguntas com 4 hipóteses. Jogo "Puzzle de Blocos": 3 níveis
	13	Quiz: 8 perguntas com 2 e 3 hipóteses. Galeria input e output: Leitura e informação simplificada.	Quiz: 8 perguntas com 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses.
	14	Quiz: 7 perguntas com 2 e 3 hipóteses. Exercício ubbox 1: As <i>skins</i> do exercício já estão selecionadas. Exercício ubbox 2: Completar o código de uma borboleta a voar e programar o Max para tirar uma fotografia. Galeria "Analogico e digital": Leitura e informação simplificada.	Quiz: 9 perguntas com 4 hipóteses. Exercício ubbox 1: As <i>skins</i> do exercício já estão selecionadas. Exercício ubbox 2: Completar o código de uma borboleta a voar e programar o Max para tirar uma fotografia. Galeria "Analogico e digital": Leitura e informação simplificada.	Quiz: 9 perguntas com 4 hipóteses. Exercício ubbox 1: As <i>skins</i> do exercício têm de ser selecionadas pelo aluno. Exercício ubbox 2: Programar uma borboleta a voar e o Max para tirar uma fotografia.
	15	Quiz: 7 perguntas com 2 e 3 hipóteses.	Quiz: 9 perguntas com 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses (1 pergunta avançada).
F. Movimentos	16	Quiz: 8 perguntas com 3 hipóteses.	Quiz: 8 perguntas com 3 e 4 hipóteses.	Quiz: 8 perguntas com 3 e 4 hipóteses.
	17	Quiz: 8 perguntas com 2 e 3 hipóteses. Galeria "Aprendizagem automática": leitura e informação simplificada.	Quiz: 8 perguntas com 3 e 4 hipóteses.	Quiz: 8 perguntas com 4 hipóteses (1 pergunta avançada).
	18	Quiz: 8 perguntas com 2 e 3 hipóteses. Exercício ubbox 1: Completar o programa de um robô a atirar água.	Quiz: 8 perguntas com 4 hipóteses. Exercício ubbox 1: Programar um robô a atirar água.	Quiz: 8 perguntas com 4 hipóteses (1 pergunta avançada). Exercício ubbox 1: Programar um robô a atirar água. Programar o som da água.
	19	Quiz: 8 perguntas com 2 e 3 hipóteses.	Quiz: 10 perguntas com 3 e 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses (uma pergunta avançada).

		Projeto ubbox: Simulação de um sismo, abordando como se deve agir sua sequência.	Projeto ubbox: Simulação de um sismo, abordando como se deve agir sua sequência.	Projeto ubbox: Simulação de um sismo, abordando como se deve agir sua sequência. Referência a placas tectónicas, incentivando a abordagem da temática de estrutura e dinâmica da Terra.
G. Movimentos, e Estilo	20	Quiz: 8 perguntas com 2 e 3 hipóteses. Jogo "Batalha naval": 3 níveis	Quiz: 9 perguntas com 3 e 4 hipóteses. Jogo "Batalha naval": 4 níveis	Quiz: 9 perguntas com 4 hipóteses. Jogo "Batalha naval": 4 níveis
	21	Quiz: 8 perguntas com 2 e 3 hipóteses.	Quiz: 9 perguntas com 3 e 4 hipóteses.	Quiz: 9 perguntas com 4 hipóteses.
	22	Quiz: 7 perguntas com 2 e 3 hipóteses. Exercício ubbox 1: Programar o Sol a pôr-se e a Lua a surgir.	Quiz: 9 perguntas com 3 e 4 hipóteses. Exercício ubbox 1: Programar o Sol a pôr-se e a Lua a surgir.	Quiz: 9 perguntas com 4 hipóteses. Exercício ubbox 1: Programar Sol a pôr-se, a Lua a surgir e um avião elétrico a aterrar.
	23	Quiz: 7 perguntas com 2 e 3 hipóteses. Projeto ubbox: Trazer a eletricidade da cidade de volta.	Quiz: 10 perguntas com 3 e 4 hipóteses. Projeto ubbox: Trazer a eletricidade da cidade de volta.	Quiz: 10 perguntas com 4 hipóteses. Projeto ubbox: Trazer a eletricidade da cidade de volta e programar as personagens a reagir. As personagens só podem andar pela estrada.
H. Controlo, Operadores e Sensores	24	Quiz: 7 perguntas com 2 e 3 hipóteses. Operações com somas e subtrações Exercícios ubbox: Operações simples com somas.	Quiz: 9 perguntas com 4 hipóteses. Operações com somas, subtrações, multiplicações e divisões. Exercícios ubbox: Operações com somas e multiplicações.	Quiz: 9 perguntas com 4 hipóteses. Operações com somas, subtrações, multiplicações e divisões. Exercícios ubbox: Operações com somas e multiplicações.
	25	Quiz: 8 perguntas com 2 e 3 hipóteses. Exercícios ubbox: Operações com somas.	Quiz: 10 perguntas com 3 e 4 hipóteses. Exercícios ubbox: Operações com somas e multiplicações.	Quiz: 10 perguntas com 4 hipóteses. Exercícios ubbox: Operações com somas e multiplicações.
	26	Quiz: 8 perguntas com 2 e 3 hipóteses. Hipóteses ilustradas.	Quiz: 10 perguntas com 3 e 4 hipóteses.	Quiz: 10 perguntas com 4 hipóteses.
	27	Quiz: 7 perguntas com 2 e 3 hipóteses. Operações com somas. Exercício ubbox 1: Programar a Ruby a apanhar mangas criando condições com 2 <i>sprites</i> . Consoante a colisão existem 2 reações diferentes.	Quiz: 6 perguntas com 4 hipóteses; 1 pergunta de verdadeiro/falso. Operações com multiplicações. Exercício ubbox 1: Programar a Ruby a apanhar mangas criando condições com 2 <i>sprites</i> . Consoante a colisão existem 2 reações diferentes.	Quiz: 6 perguntas com 4 hipóteses; 1 pergunta de verdadeiro/falso. Operações com multiplicações. Exercício ubbox 1: Programar a Ruby a apanhar mangas criando condições com 4 <i>sprites</i> . Consoante a colisão existem 5 reações diferentes.
	28	Quiz: 7 perguntas com 2 e 3 hipóteses.	Quiz: 7 perguntas com 4 hipóteses.	Quiz: 7 perguntas com 4 hipóteses (1 pergunta avançada).
I. Projetos Finais	29	Projeto ubbox: Abordagem de ações que prejudicam o ambiente.	Projeto ubbox: Abordagem de ações que prejudicam o ambiente.	Projeto ubbox: Abordagem de ações que prejudicam o ambiente. Diferença e exemplos de energias renováveis e não renováveis.
	30	Sem variantes.	Sem variantes.	Sem variantes.