

---

**Critérios de Avaliação da Área Disciplinar de Informática**  
**Ano Letivo 2017/2018**

**Iniciação à Programação no 1.º Ciclo do Ensino Básico**  
**3.º e 4.º Anos**

**Avaliação**

O projeto Iniciação à Programação no 1.º Ciclo do Ensino Básico pode ser implementado numa grande diversidade de cenários, é importante que sejam pensados à partida os mecanismos de avaliação dos alunos de modo a respeitar as opções globais definidas pelo projeto do agrupamento. Privilegia-se uma avaliação baseada em evidências recolhidas ao longo das sessões em que os alunos desenvolvem os seus projetos. Para tal serão utilizadas grelhas de observação, tomadas notas, bem como recolhidos registos ao longo das sessões que permitam ficar com evidências do processo de criação dos projetos por parte dos alunos. Outros elementos a ter em linha de conta serão os produtos elaborados pelos alunos e o modo como os apresentam e documentam. Deverá ficar claro, no entanto, que, mais que o produto, interessa o processo e o modo como cada grupo trabalhou para chegar a esse produto.

Não se considera apropriada a utilização de testes práticos ou teóricos bem como fichas escritas individuais como forma de avaliação dos conhecimentos no âmbito deste projeto, uma vez que mais do que a memorização de comandos e outras formas que estes testes poderiam medir se pretende aferir também a criatividade, o empenho e a compreensão de um modo geral dos conceitos subjacentes ao pensamento computacional.

Para a avaliação serão usadas as grelhas propostas por Brenann, Balsch, e Chung (2014), traduzidas por Ramos e Espadeiro (2015):

TESTAR E CORRIGIR	BAIXO	MÉDIO	ALTO - ELEVADO
Descreve o que aconteceu com o projeto de diferente em relação ao pretendido.	Não descreve o que resultou diferente em relação ao pretendido.	Descreve o que correu mal no projeto, mas não o que pretendia fazer.	Dá um exemplo detalhado do que aconteceu e o que pretendia, quando executa o programa.
Descreve de que forma foi feita a leitura do código para encontrar a causa do problema.	Não descreve um problema.	Descreve como faz a leitura mas não apresenta um exemplo específico de encontrar um problema no código.	Descreve como faz a leitura e apresenta um exemplo específico de encontrar um problema no código
Descreve que alterações fez e como as testou para verificar os resultados.	Não descreve que problemas teve ou a solução	Fornece um exemplo genérico sobre as alterações feitas e os testes feitos para verificar o funcionamento.	Fornece um exemplo específico sobre as alterações feitas e os testes feitos para verificar o funcionamento.
Descreve outras formas de resolver o problema.	Não apresenta uma forma para encontrar uma solução para o problema.	Apresenta uma forma genérica para encontrar uma solução para o problema.	Apresenta um exemplo específico de como encontrar uma solução para o problema.

EXPERIMENTAR E INTERAGIR	BAIXO	MÉDIO	ALTO - ELEVADO
Descreve a construção do projeto, passo a passo.	Fornecer uma descrição elementar da construção do projeto, mas não detalha aspetos específicos do mesmo.	Faz uma descrição genérica do projeto, de forma ordenada.	Fornecer detalhes sobre as diferentes componentes dum projeto específico e descreve o modo como foram desenvolvidos, de forma ordenada.
Enumera exemplos que experimentou ao longo da elaboração do projeto?	Não apresenta exemplos específicos do que experimentou.	Deixa transparecer de forma genérica que experimentou outras coisas no projeto.	Fornecer exemplos específicos de outras coisas que foi experimentando no projeto.
Apresenta as revisões que foram sendo feitas.	Afirma não ter feito revisões, ou afirma ter feito algumas mas não exemplifica.	Descreve uma revisão específica que fez ao projeto.	Descreve aspetos específicos de coisas que acrescentou ao projeto e justifica.
Descreve as diferentes abordagens que experimentou no projeto, ou quando tentou fazer algo novo.	Não revela evidências de ter experimentado algo novo.	Fornecer um exemplo de algo novo que experimentou no projeto.	Descreve com detalhe coisas novas que experimentou no projeto.

REUTILIZAR E RECOMBINAR	BAIXO	MÉDIO	ALTO
Descreve se encontrou inspiração em outros projetos e na leitura do código disponível.	Descreve como desenvolveu as ideias ou em que projetos se inspirou.	Fornecer uma descrição geral de um projeto que o inspirou.	Dá um exemplo específico do projeto que o/a inspirou.
Descreve como selecionou uma parte de outro projeto e como a adaptou ao seu.	Não descreve como adaptou as ideias, <i>scripts</i> ou recursos de outros projetos.	Identifica <i>scripts</i> , ideias ou recursos que adaptou de outros projetos.	Fornecer exemplos específicos de <i>scripts</i> , ideias ou recursos que ele adaptou de outros projetos e como.
Como refere/cita as pessoas cujo trabalho inspirou o seu.	Não identifica as fontes e os autores em que se inspirou para a realização do seu projeto.	Identifica as fontes e os autores em que se inspirou para a realização do seu projeto.	Documenta no projeto as fontes e os autores que o inspiraram

ABSTRAIR E MODULARIZAR	BAIXO	MÉDIO	ALTO
Como foi decidido que <i>sprites</i> eram necessários para o projeto e onde eram utilizados.	Não descreve que <i>sprites</i> foram selecionados.	Fornecer uma descrição geral da decisão de escolher certos <i>sprites</i> .	Dá uma explicação detalhada acerca de como selecionou os <i>sprites</i> em função do objetivo do projeto.
Como foi decidido que <i>scripts</i> eram necessários para o projeto e onde eram utilizados.	Não descreve que <i>scripts</i> foram criados.	Fornecer uma descrição geral da decisão de criar certos <i>scripts</i> .	Dá uma explicação detalhada acerca de como criou os <i>scripts</i> em função do objetivo do projeto.
Como foram organizados os <i>scripts</i> de forma a terem significado para o estudante e para os outros.	Não descreve como os <i>scripts</i> foram organizados.	Fornecer uma descrição geral da forma como foram organizados <i>scripts</i> .	Dá uma explicação detalhada acerca de como organizou os <i>scripts</i> e porq

A classificação a atribuir no final de cada período será de acordo com a seguinte escala:

Quantitativo	Qualitativo
1	Muito Insuficiente

2	Insuficiente
3	Suficiente
4	Bom
5	Muito Bom

E será obtida tendo em conta o patamar atingido em cada um dos parâmetros

- Testar e corrigir;
- Experimentar e interagir;
- Reutilizar e recombina;
- Abstrair e modularizar;

de acordo com a seguinte combinação de parâmetros:

Quantitativo	Qualitativo	Baixo	Médio	Alto
1	Muito Insuficiente	3 ou 4	0 ou 1	0 ou 1
2	Insuficiente	2	1 ou 2	0 ou 1
3	Suficiente	0 ou 1	2,3 ou 4	0 ou 1
4	Bom	0	2 ou 3	1 ou 2
5	Muito Bom	0	0 ou 1	3 ou 4

**Nota:** Na avaliação serão sempre tidas em conta a progressão e regressão dos alunos, ao longo do ano letivo, sendo no final do ano atribuída ao aluno uma classificação que traduz o trabalho desenvolvido.

#### Bibliografia:

Brennan, K., Balch, C. & Chung, M. (2014) *Creative computing*. Harvard Graduate School of Education. Disponível em:  
<http://scratched.gse.harvard.edu/guide/files/CreativeComputing20141015.pdf>. Acedido em agosto de 2015.

Ramos, J.L. & Espadeiro, R.G. (2015) *Pensamento computacional na escola e práticas de avaliação das aprendizagens. Uma revisão sistemática da literatura*. Disponível em:  
<http://dspace.uevora.pt/rdpc/bitstream/10174/14227/1/challenges%202015br.pdf>. Acedido em agosto 2015.