

 <p>UNIVERSIDADE DE COIMBRA FACULDADE DE CIÊNCIAS E TECNOLOGIA</p> <p><i>Departamento de Engenharia Informática</i></p>	<p>Projeto #1 Algoritmos e Estruturas de Dados</p> <p>2021-2022</p> <p>Texto de Apoio – Como usar o <i>Mooshak</i></p>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------

Objetivos:

Este texto de apoio, enquadrado no projeto #1 de AED, tem por objetivo dar apoio aos alunos que ainda não tenham tido qualquer experiência com o *Mooshak*, a plataforma de submissão de problemas de programação que será usada nos diversos projetos da disciplina.

O que é o *Mooshak*?

O *Mooshak* é uma plataforma para gerir concursos de programação, desenvolvida pela Universidade do Porto. Para além disso, tem sido também usada para dar apoio a disciplinas de programação. É nesse contexto que o *Mooshak* será usado em AED.

Como é usado o *Mooshak* em AED:

Foi instalado um servidor *Mooshak* de suporte às aulas de AED (<https://mooshak.dei.uc.pt/~aed>) e, usando esse servidor, os projetos funcionarão da seguinte forma:

- Para cada projeto os docentes criam um “concurso” (*contest*).
- Cada concurso pode ter vários problemas, cada um deles com um enunciado específico e com prazos de submissão específicos. O projeto pode ser assim constituído por várias metas (por exemplo: *problemas A e B entregues obrigatoriamente antes do final da aula prática a definir*).
- Os alunos, usando as contas previamente criadas, submetem no *Mooshak* o código-fonte da sua solução para um problema. O *Mooshak* compila e testa esse código, usando um conjunto de *inputs* de teste previamente definido pelos docentes, e reporta se o código funciona de acordo com as especificações do enunciado ou não.
- Os alunos podem fazer sucessivas submissões e visualizar todas as submissões efetuadas.
- Os alunos poderão usar o *Mooshak* para colocar questões aos docentes (dúvidas sobre o enunciado ou sobre o projeto, etc.), que responderão usando também o *Mooshak*. Espera-se que este seja o principal canal de suporte aos projetos (para além das próprias aulas).
- Os docentes podem verificar, no *Mooshak*, o histórico de submissões de cada aluno, o respetivo código-fonte, e o resultado dos testes feitos pelo *Mooshak*. Esses elementos ajudam os docentes a avaliar o projeto, para além da avaliação resultante da interação do docente com os alunos no espaço da sala de aula.

Deve, no entanto, ser referido que o *Mooshak* é apenas um dos instrumentos de avaliação dos projetos, complementando a interação dos docentes com o aluno e os relatórios escritos, nos casos em que estes integrem o projeto.

Ter o código aceite pelo *Mooshak* não é por si só garantia de uma boa classificação, e não conseguir que o *Mooshak* aceite o código-fonte submetido indicia obviamente alguns problemas na implementação ou interpretação do enunciado mas não implica necessariamente uma avaliação negativa do projeto do aluno.

Como posso obter uma conta no *Mooshak*?

Todos os alunos devem registar-se no *Mooshak* de AED. Para isso deverão:

1. Aceder à página principal (<https://mooshak.dei.uc.pt/~aed>)
2. Seguir a opção “Register for on-line contest”
3. Preencher o formulário:
Contest: **AED Projeto 1**
Name: **número de estudante (e.g. 2019000000)**
Email: **endereço de e-mail do DEI ou da UC (e.g. exemplo@student.dei.uc.pt)**
Group: **AED**
4. A *password* é enviada para a caixa de correio do e-mail definido
5. Usar essa *password* para fazer *login* no concurso AED Projeto 1

Para os projetos (*contests*) seguintes este processo é desnecessário, pois as contas serão automaticamente copiadas.

Como verifica o *Mooshak* as minhas submissões?

Usando os menus o aluno submete no *Mooshak* o seu código fonte (.c, .cpp, .java, ou .py) que será compilado e executado pelo *Mooshak* usando um conjunto de *inputs* previamente definidos pelos docentes. A execução do código submetido deve produzir, para cada *input*, um *output* exatamente igual ao que foi definido pelo docente (incluindo a formatação do *output*).

Os docentes definem vários pares *input/output* para poder testar todas as situações de funcionamento (casos “normais”, casos “extremos”, situações de exceção, etc.) e os alunos não têm acesso aos *inputs* usados pelo *Mooshak*, para evitar situações de fraude e, principalmente, para estimular uma análise mais cuidada dos enunciados e do código-fonte desenvolvido.

As respostas mais habituais do *Mooshak* às submissões dos alunos são as seguintes:

- **Accepted** – o programa devolve o *output* esperado em todos os testes definidos pelos docentes.
- **Compile Time Error** – o *Mooshak* não conseguiu compilar o código porque foi gerado um erro ou aviso durante a compilação. Ao carregar no texto “Compile Time Error” é possível visualizar o erro de compilação que ocorreu.
- **Runtime Error** – a execução do programa terminou prematuramente. Isto pode ser causado por um *segmentation fault* ou por uma *exception*, por exemplo.
- **Time Limit Exceeded** – o programa não terminou dentro do limite de tempo definido. O programa pode estar por exemplo “preso” num ciclo infinito ou ter um algoritmo excessivamente ineficiente, que penaliza de forma notória o tempo de execução e que deve ser otimizado.
- **Wrong Answer** – o *output* do programa não é igual ao esperado, pelo menos para um dos testes efectuados. Isto significa que o programa produz *outputs* errados ou cuja formatação não está de acordo com o definido no enunciado.
- **Presentation Error** – o programa produz *outputs* corretos mas com problemas de formatação (espaços a mais, por exemplo). O *Mooshak* nem sempre consegue distinguir entre “Presentation Error” e “Wrong Answer”, sendo que nesses casos reporta “Wrong Answer”.

- **Invalid Submission** – o ficheiro do código-fonte tem uma extensão inválida ou correspondente a uma linguagem de programação não suportada.

Como deve o meu programa ler o *input* e produzir o *output*?

O *Mooshak* espera que os programas leiam o *input* da entrada *default* (*stdin*) e que produzam o *output* para a saída *default* (*stdout*).

Um erro típico é o uso de menus interativos (por exemplo “Digite o número primo a converter: ”) ou leitura explícita de ficheiros no código submetido (por exemplo “Indique o nome do ficheiro de *input*: ”), que não são aceites pelo *Mooshak*.

Ao testar o código antes de o submeter no *Mooshak* podem:

- Introduzir o *input* manualmente na linha de comandos ou IDE, nos casos em que a dimensão do *input* seja reduzida.
- Redirecionar o *input* de um ficheiro para o *stdin* com um comando do tipo:

```
java HelloWorld < fichInput.txt
```

 Notem que também podem redirecionar o *stdout* para um ficheiro:

```
java HelloWorld < fichInput.txt > fichOutput.txt
```
- Ou, no IDE que estejam a usar, definir um ficheiro de texto como fonte *stdin*.

Que linguagens de programação usar?

O *Mooshak* suporta várias linguagens de programação (C, C++, Java, e *Python 3*). Os comandos usados para compilar e/ou executar o código fonte são:

Extensão	Compilador	Versão	Command
.c	gcc	8.3.1	gcc -std=c11 -Wall -Wextra -O3 "\$file" -lm
.cpp	g++	8.3.1	g++ -std=c++17 -Wall -Wextra -O3 "\$file" -lm
.java	javac (OpenJDK)	11.0.9.1	javac "\$file" java -Xss8m "\$name"
.py	pypy3	3.7.9	pypy3 "\$file"

Nos projetos, salvo indicações em contrário por parte dos docentes, os alunos são livres de escolher a linguagem de programação usada na resolução do problema, dado que a disciplina se foca essencialmente na algoritmia. No entanto, é importante notar que a qualidade do apoio prestado pelos docentes aos alunos poderá apresentar variações, consoante a maior ou menor experiência do docente com a linguagem que o aluno esteja a usar.