**Guião final**

**Plágio**

Ao colocar o código num repositório público, é sempre possível que alguém possa copiar o código de outro projeto. De acordo com o código de ética da Universidade do Minho, tal ação é considerada plágio. Caso haja dúvida em relação a plágio, a equipa docente irá utilizar informação sobre os *commits* para decidir quem produziu o código e quem o copiou. Caso a equipa docente decida que houve plágio, os grupos que sejam considerados autores de plágio terão **ZERO** na componente de projeto.

**Avaliação da componente de grupo**

A avaliação da componente de grupo é feita da seguinte forma:

* Cada um dos guiões 5, 6, 7, 8, 9 e 10 vale 1 ponto
* A avaliação final vale 6 pontos
* No máximo 2 pontos serão atribuídos pela participação no torneio

**Avaliação final**

A avaliação final será feita utilizando os seguintes critérios:

* Documentação (2 pontos)
* Modularidade e legibilidade (2 pontos)
* Deteção dos fins de jogo (1 ponto)
* Compilar sem warnings (1 ponto)

**Documentação**

Todas as funções do vosso projeto, assim como estruturas de dados e macros deverão ser documentadas. A percertagem da documentação será utilizada para a avaliação deste critério.

**Modularidade e legibilidade**

Pretende-se que o código esteja bem escrito e legível. Isto implica:

* Não escrever código repetitivo
* Nao replicar código
* Não utilizar funções demasiado compridas (deverão caber num ecrã)
* Utilizar funções auxiliares com nomes sugestivos que ilustrem o que fazem
* As variáveis deverão ter nomes sugestivos
* Não aceder diretamente aos dados
* Estruturar o código em módulos estanques
* Não incluir código
* Não utilizar variáveis globais
* Não misturar a camada de interface com o resto do código
* Não misturar os dados com o resto do código

**Deteção dos fins de jogo**

O vosso projeto deverá detetar todas as instâncias de fim de jogo.

**Compilar sem warnings**

Pretende-se que o seu programa compile **sem qualquer warning** ao compilar utilizando as seguintes opções:

-std=gnu11 -Wall -Wextra -pedantic-errors -O

A título de exemplo, se todo o vosso código estiver numa única pasta e essa pasta não contiver outros ficheiros C, o seguinte comando

gcc -std=gnu11 -Wall -Wextra -pedantic-errors -O \*.c

deveria compilar o vosso programa e gerar o executável **a.out** sem mostrar *warnings*. A avaliação deste critério é binária:

* **100%** da cotação se a compilação ocorreu sem qualquer warning
* **0%** da cotação se ocorreu pelo menos um warning ao compilar

**Bot**

A título de bonificação, os grupos que quiserem deverão submeter o seu programa para um torneio. Cada programa deverá:

1. Ler o estado do jogo a partir de um ficheiro. O nome desse ficheiro deverá ser passado como parâmetro ao programa;
2. Efetuar a melhor jogada no menor tempo possível (**2 segundos de CPU**);
3. Gravar o estado do jogo num ficheiro. O nome desse ficheiro deverá ser passado como parâmetro ao programa.

Assim, se o programa for invocado da seguinte forma:

./bot jog01 jog02

Então ele deverá ler o estado do ficheiro **jog01**, jogar e gravar o estado no ficheiro **jog02**.

**Avaliação do bot**

A competição será através de um campeonato. Este proceder-se-á da seguinte forma:

* Cada jogador jogará contra todos os outros duas vezes, uma como primeiro jogador e uma como segundo jogador
* Um jogador perde o jogo se criar uma jogada for inválida, porque:
  + demorou mais do que o tempo permitido, ou
  + o formato do tabuleiro não está correto, ou
  + o tabuleiro não corresponde a uma jogada válida
* Cada jogo ganho vale 1 ponto
* Após todos os jogadores terem jogado contra os outros, ordenam-se pela pontuação obtida
* Caso vários grupos obtenham a mesma pontuação, eles partilharão a mesma posição

Cada grupo receberá:

| **Posição** | **Avaliação** |
| --- | --- |
| 1º lugar | 2.00 pontos |
| 2º lugar | 1.75 pontos |
| 3º e 4º lugares | 1.50 pontos |
| 5º ao 8º lugares | 1.25 pontos |
| 9º ao 16º lugares | 1.00 pontos |
| Derrotar o jogador aleatório | 0.50 pontos |

**Software**

O software utilizado na avaliação está em <https://github.com/equipadocente-la1li21920/Torneio>. O desenvolvimento ainda não acabou mas pensamos que já deve poder ser utilizado para experimentar (e reportar bugs, claro).

**Entrega**

O projeto só será **aceite** se se respeitarem as regras descritas abaixo. A entrega será feita até ao dia **3 de Maio** de duas formas (ambas **obrigatórias**):

* através dos commits efetuados no **Github**
* através de um arquivo ZIP (para arquivo)

As seguintes regras terão que ser respeitadas: O arquivo e o projeto no Github deverão **obrigatoriamente** ter as seguintes pastas e ficheiros na sua raiz:

* ficheiro README.md que deverá conter:
  + o nome do curso, LCC ou MIEI
  + o nome do turno, PL, e.g., PL1
  + o número de aluno de cada elemento seguido do seu nome completo
* pasta projeto contendo todo o código do projeto
* pasta bot contendo todo o código correspondente ao bot (caso pretenda ser avaliado no torneio)
* pasta doc contendo toda a documentação do projeto (gerada utilizando o Doxygen) em formato XML

O nome do arquivo submetido no *Blackboard* deverá ter o seguinte formato:

<nome da UC>PL<número do turno prático>G<numéro do grupo com dois algarismos>.zip

O número nos casos dos grupos com um só algarismo é obtido colocando um zero como *padding*. Assim:

la1PL2G03.zip

Será o projeto do grupo 3 do turno PL2 de Laboratório de Algoritmia I

li2PL4G09.zip

Será o projeto do grupo 9 do turno PL4 de Laboratórios de Informática II

Esse arquivo deverá ser submetido por um dos elementos do grupo na plataforma de Elearning (vulgo Blackboard) até ao dia **3 de Maio**.

**Entrega do Bot**

A entrega do bot deverá seguir exatamente as mesmas regras mas ser feita num link para a submissão dos bots na plataforma blackboard até ao dia **3 de Maio**.

**Defesa**

As defesas ocorrerão na semana de **11 a 15 de Maio** para permitirem aos docentes avaliar os projetos e correr o torneio. As defesas decorrerão da seguinte forma:

* Cada grupo marcará um período na semana de 11 a 15 de Maio para a avaliação num link que será disponibilizado para esse efeito
* A defesa será no Blackboard Collaborate Ultra
* Estarão presentes dois docentes em cada defesa
* A defesa durará no máximo 20 minutos
* Para a defesa é necessário que os alunos possam utilizar o vídeo e o áudio
* Todos os elementos terão que comparecer à defesa para serem avaliados
* Serão feitas perguntas a cada um dos elementos do grupo para esclarecer dúvidas dos docentes sobre o trabalho, o seu funcionamento e a participação de cada um dos elementos na sua elaboração
* A avaliação de cada um dos elementos do grupo poderá ser diferente