

UNIVERSIDADE AUTÓNOMA DE LISBOA LUÍS DE CAMÕES
Departamento de Engenharias e Ciências da Computação
Algoritmia e Programação 2022/2023
Projeto

"N" em linha

1 Datas Relevantes

Data	Evento
05/12/2022	Disponibilização do enunciado.
13/01/2023 23:59:59 GMT	Entrega final do trabalho.
16/01/2023 — 27/01/2023	Apresentação dos trabalhos.

Tabela 1: Data relevantes

2 Descrição

Pretende-se a implementação de um jogo semelhante ao 4 em linha, como ilustrado pela Figura 1. A principal diferença está no número variável de peças, i.e., que não está limitado a 4.

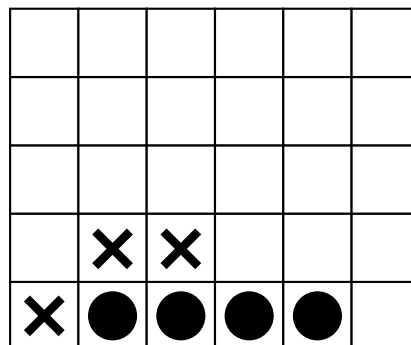


Figura 1: Exemplo de jogo terminado, com tamanho de sequência vencedora igual a 4.

O jogo decorre numa grelha vertical, onde são colocadas peças pelo topo. Cada peça desce até à base da grelha, ou até ao topo da última peça colocada nessa posição.

Além do número variável de peças necessárias para ganhar o jogo, cada jogador dispõe de um número limitado de peças especiais, i.e., sequências de peças com várias dimensões. Embora no jogo tradicional só seja possível colocar uma peça de cada vez, nesta versão será possível colocar estas sequências.

Deve ser possível registar jogadores, e registar o número vitórias e jogos jogados de cada um.

Para aprovação em Algoritmia e Programação, o trabalho tem que implementar as instruções descritas na Secção 3. Estas descrevem um conjunto de instruções com as quais os jogadores, através do terminal, interagem com o programa.

2.1 Colocação de peças

O jogo original permite colocar apenas uma peça de cada vez, até um dos jogadores conseguir uma sequência vertical, horizontal, ou diagonal de peças com dimensão 4 (como ilustrado na Figura 2. Neste

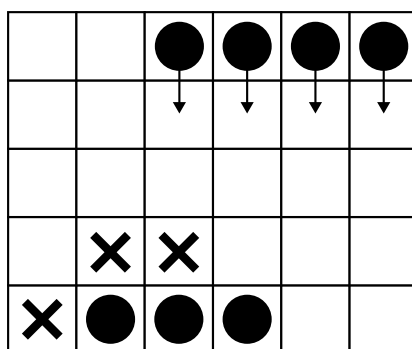


Figura 2: Colocação de peça especial com tamanho 4, na posição 3 D, ou 6 E.

jogo, além de ser possível variar o número de peças necessárias para vencer (sequências vencedora), vai ser possível colocar também várias peças de uma só vez, através das peças especiais.

Cada jogador tem disponíveis peças unitárias ilimitadas, e um conjunto de sequências de peças, i.e., peças especiais. A colocação de peças especiais é feita indicando a posição inicial, e o sentido (i.e., esquerda (E) ou direita(D)). As peças descem na grelha e ficam colocadas de acordo com a ocupação atual (ver Figura 3).

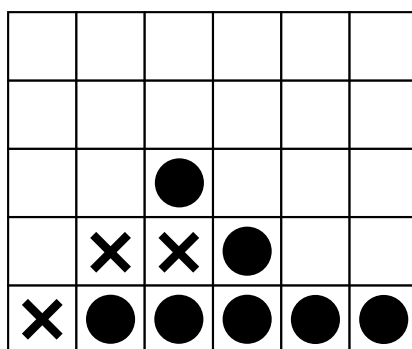


Figura 3: Resultado da colocação de peça especial com tamanho 4, na posição 3 D, ou 6 E.

2.2 Parâmetros de jogo

Um jogo opõe dois jogadores registrados, e tem os seguintes parâmetros:

- w — Comprimento da grelha, em peças, onde $w \in \mathbb{N}$;
- h — Altura da grelha em peças, onde $h \in \mathbb{N}, \lceil \frac{w}{2} \rceil \leq h \leq w$;
- n — Número de peças em linha para determinar a vitória (i.e., TamanhoSequência), onde $n \in \mathbb{N}, n \leq w$;
- S — Conjunto de tamanhos de peças especiais disponível para cada jogador, onde $\forall s \in S : S \in \mathbb{N}, s < n$. E.g., $n = 5, S = [3, 3, 4]$, implica que cada jogador dispõe de duas peças especiais de 3 peças, e uma peça especial de 4 peças, além do número ilimitado de peças unitárias.

3 Instruções

Na descrição das várias instruções é indicada a sua sintaxe. Os argumentos são separados por espaços em branco, e cada linha é terminada por um carácter fim de linha.

De forma a oferecer precisão às descrições das instruções, os símbolos invisíveis são representados graficamente, i.e., espaços em branco por , e finais de linha por `\n`. Na implementação devem ser utilizados os símbolos invisíveis correspondentes.

Para cada instrução é indicada a lista de expressões de saída, quer para execuções com sucesso, quer para insucesso.

No caso de insucesso só deve surgir uma mensagem de erro. Verificando-se várias situações de insucesso em simultâneo, deve surgir apenas a mensagem do primeiro cenário, de acordo com a ordem de saídas de insucesso descritas para cada instrução.

Caso o utilizador introduza uma instrução inválida, ou seja, não prevista na lista de instruções desta secção, ou um número de parâmetros errado para uma instrução existente, o programa deve escrever:

```
Instrução inválida.
```

A descrição de cada instrução pretende ser exaustiva, e sem ambiguidades. Não deve ser possível optar entre vários comportamentos possíveis na mesma situação. Se essa situação ocorrer deve entrar em contacto com a docente.

A implementação não deve suportar mais instruções do que as que estão descritas.

Os parâmetros `Nome` e `Tipo` correspondem ao nome de jogadores, e tipo de peça.

O sucesso da maioria das instruções depende da existência de um jogo em curso, sendo que esse estado deve ser verificado. Note-se que um jogo em curso opõe dois jogadores, e só esses jogadores podem colocar peças.

Registar jogador (RJ) Regista um novo jogador. Cada jogador tem um nome único. Não existe limite para o número de jogadores registados.

Nome é um nome de um jogador.

Entrada:

```
RJ Nome
```

Saída com sucesso:

```
Jogador registado com sucesso.
```

Saída com insucesso:

- Quando o jogador indicado já se encontra registado.
`Jogador existente.`

Remover jogador (EJ) Remove um jogador previamente registado, se este não estiver a participar no jogo em curso.

Nome é um nome de um jogador.

Entrada:

```
EJ Nome
```

Saída com sucesso:

```
Jogador removido com sucesso.
```

Saída com insucesso:

- Quando o jogador indicado não se encontra registado.
`Jogador não existente.`

- Quando o jogador indicado participa no jogo em curso.
Jogador participa no jogo em curso.

Listar jogadores (LJ) Lista os jogadores registados por ordem alfabética, indicando o número de jogos jogados e vitórias.

Nome é um nome de um jogador, Jogos é o número de jogos jogados pelo jogador, e Vitórias é o seu número de vitórias.

Entrada:

```
LJ
```

Saída com sucesso:

```
Nome Jogos Vitorias
Nome Jogos Vitorias
...
```

Saída com insucesso:

- Quando não existem jogadores registados.
Não existem jogadores registados.

Iniciar jogo (IJ) Inicia um novo jogo, se não estiver ainda um jogo iniciado. É necessário indicar o nome dos jogadores que irão participar, as dimensões da grelha, o tamanho da sequência vencedora, e as peças especiais disponíveis. Os jogadores têm que estar previamente registados. O jogo inicia indicando o nome dos jogadores que nele participam por ordem alfabética.

Nome é um nome de um jogador, Comprimento e Altura são as dimensões da grelha de jogo, TamanhoSequência é o tamanho da sequência vencedora, e TamanhoPeça é o tamanho de uma peça especial.

Nenhum valor de TamanhoPeça pode ser igual ou superior a TamanhoSequência (ver secção 2.2).

Entrada:

```
IJ Nome Nome
Comprimento Altura TamanhoSequência
TamanhoPeça TamanhoPeça ... TamanhoPeça
```

Saída com sucesso:

```
Jogo iniciado entre Nome e Nome.
```

Saída com insucesso:

- Quando já existe um jogo em curso.
Existe um jogo em curso.
- Quando algum dos jogadores indicados não se encontra registado.
Jogador não registado.
- Quando as dimensões da grelha não respeitam as regras indicadas na secção 2.2.
Dimensões de grelha invalidas.
- Quando tamanho da sequência vencedora não respeita as regras indicadas na Secção 2.2.
Tamanho de sequência invalido.
- Quando alguma dimensão das peças especiais não respeita as regras indicadas na Secção 2.2.
Dimensões de peças especiais invalidas.

Detalhes de jogo (DJ) Mostra informação sobre o jogo em curso: tamanho da grelha, jogadores por ordem alfabética, e tipo e quantidade de peças especiais disponíveis para cada jogador.

Comprimento e Altura são as dimensões da grelha de jogo, Nome é um nome de um jogador, TamanhoPeça é o tamanho de uma peça especial, e Quantidade é o número de peças especiais atualmente disponíveis para o jogador.

Entrada:

```
DJ
```

Saída com sucesso:

```
Comprimento  Altura
Nome
TamanhoPeça  Quantidade
TamanhoPeça  Quantidade
...
Nome
TamanhoPeça  Quantidade
TamanhoPeça  Quantidade
...
```

Saída com insucesso:

- Quando não existe jogo em curso.
Não existe jogo em curso.

Desistir (D) Um jogador pode desistir do jogo em curso do qual faz parte, pelo que o outro jogador regista uma vitória. Ambos registam mais um jogo jogado. Caso os dois jogadores desistam, ambos somam um jogo jogado, sem vitória atribuída.

Nome é um nome de um jogador. O segundo parâmetro Nome é opcional.

Entrada:

```
D Nome[ Nome]
```

Saída com sucesso:

```
Desistência com sucesso. Jogo terminado.
```

Saída com insucesso:

- Quando algum dos jogadores indicados não se encontra registado.
Jogador não registado.
- Quando não existe jogo em curso.
Não existe jogo em curso.
- Quando um dos jogadores indicados não participa no jogo em curso.
Jogador não participa no jogo em curso.

Colocar peça (CP) Um jogador coloca uma peça, se fizer parte do jogo em curso, e for a sua vez. Se nenhum jogador tiver colocado peças no jogo em curso, qualquer um pode ser o primeiro.

Nome é um nome de um jogador, TamanhoPeça é o tamanho da peça a colocar, Posição é a coordenada horizontal onde a peça será colocada, e Sentido é o sentido (E ou D), na horizontal, para onde a peça será colocada (ver Secção 2.1). Sentido é opcional, podendo ser omitido quando a peça tem tamanho 1.

Entrada:

CP Nome TamanhoPeça Posição[Sento]

Saída com sucesso:

- Quando a peça colocada possibilita uma sequência vencedora.
Sequência conseguida. Jogo terminado.
- Após colocação da peça.
Peça colocada.

Saída com insucesso:

- Quando não existe jogo em curso.
Não existe jogo em curso.
- Quando o jogador indicado não participa no jogo em curso.
Jogador não participa no jogo em curso.
- Quando não for possível utilizar uma peça com o tamanho indicado (por inexistência ou indisponibilidade).
Tamanho de peça não disponível.
- Quando a posição e sentido indicados não permitirem a colocação da peça, de acordo com as regras indicadas na Secção 2.2.
Posição irregular.

Visualizar resultado (V) Mostra o estado atual da grelha do jogo em curso, indicando o conteúdo de cada posição. As posições são mostradas de cima para baixo, da esquerda para a direita, indicando linha, coluna, e conteúdo.

	1	2	3	4	5	6
1						
2						
3						
4						
5						

Figura 4: Tabuleiro de jogo vazio

Conteúdo pode ser Vazio, caso nenhuma peça se encontra na posição, ou o nome do jogador com peça na posição. Altura e Comprimento são as dimensões da grelha.

Entrada:

V

Saída com sucesso:

```
1 1 Conteudo
1 2 Conteudo
...
Altura Comprimento Conteudo
```

Saída com insucesso:

- Quando não existe jogo em curso.
Não existe jogo em curso.

Gravar (G) Grava a lista de jogadores, resultados, e estado do jogo em curso (se existir) em ficheiro.

NomeFicheiro é o nome do ficheiro onde será feita a gravação.

Entrada:

G NomeFicheiro

Saída com sucesso:

Jogo gravado.

Saída com insucesso:

- No caso de ocorrer um erro de escrita.
Ocorreu um erro na gravação.

Ler (L) Lê a lista de jogadores, resultados, e estado do jogo em curso (se existir) de ficheiro.

NomeFicheiro é o nome do ficheiro de onde será feita a leitura.

Entrada:

L NomeFicheiro

Saída com sucesso:

Jogo carregado.

Saída com insucesso:

- No caso de ocorrer um erro de acesso ao ficheiro, ou dados corrompidos.
Ocorreu um erro no carregamento.

3.1 Entrega

A entrega do projeto é feita na plataforma de *e-learning* e deve ser submetido:

- Um ficheiro zip com o código completo da implementação. Deve existir, na raiz do zip, um ficheiro chamado `program.py`.

3.2 Apresentação

Todos os projetos entregues serão sujeitos a uma apresentação (0.20). Para esse efeito, cada grupo terá que efetuar uma apresentação do projeto desenvolvido para turma. De forma a demonstrar que o código entregue foi de facto feito pelo grupo, e que a distribuição de trabalho foi equilibrada.

O calendário das apresentações será disponibilizado no *e-learning*, após o prazo de entrega da implementação do projeto. Cada apresentação tem a duração máxima de 15 minutos. Cada apresentação deve, pelo menos, focar os seguintes pontos:

- **Introdução:** Apresentação breve do projeto e respetivo objetivo.
- **Benchmark:** Indicação de alguns projetos semelhantes já implementados em Python e disponíveis. Sugestão: Procure nos códigos disponibilizados no GitHub.
- **Estrutura do projeto:** Apresentação da organização do projeto e respetivas estruturas utilizadas.
- **Simulação de jogo:** Apresentação de um exemplo do jogo em funcionamento, incluído todas as funcionalidades implementadas.
- **Conclusões.**

Como auxílio deve desenvolver um PPT.

A não comparência na apresentação implica a anulação da participação individual no projeto.

3.3 Avaliação

O projeto é avaliado com base em duas componentes: quantitativa (A); e qualitativa (B). A tabela 2 contém as valorizações quantitativas de cada instrução.

Instrução	Cotação
RJ	1 valor
EJ	1 valor
LJ	2 valor
IJ	2 valor
DJ	3 valores
D	2 valores
CP	4 valores
V	3 valores
G	1 valor
L	1 valor

Tabela 2: Avaliação das funcionalidades do Projeto

A nota final do projeto é determinada por: $(0.5 \times A) + (0.3 \times B) + (0.2 \times Apresentacao)$

A avaliação qualitativa irá considerar que existem várias formas de resolver o problema descrito, mas que se exige a utilização dos instrumentos e métodos apresentados na unidade curricular de Algoritmia e Programação, nomeadamente:

- Separação entre interface, dados, e lógica da aplicação (modelo MVC);
- Justificação clara para as variáveis e operações implementadas;
- Organização da solução coerente com a metodologia apresentada na disciplina;
- Adequação da escolha de estruturas de dados e algoritmos para a resolução do problema.

As notas finais do projeto serão disponibilizadas no *e-learning*.