

	<b>Segundo Exercício Escolar</b>	Semestre: <b>2020.3</b>	<b>12/11/2019</b>
	Curso: Sistemas de Informações	Disciplina: LPA I	
	Prof: Jackson Raniel Florencio da Silva		<b>Nota:</b>
	Aluno:		
Orientações para a avaliação			
<b>Obs.:</b> A prova é individual e sem consulta. O arquivo de extensão .py com o programa desenvolvido deve ser anexado como reposta no Google Classroom. O descumprimento de qualquer desses termos será penalizado com a atribuição de nota 0 ao resultado dessa avaliação.			

## Pisando em ovos

Você foi contratado para desenvolver o jogo que revolucionará o mercado de jogos baseados em texto. De nome “Pisando em ovos”, esse jogo é uma divertida mistura entre campo minado e a caça aos ovos de páscoa. Uma aventura eletrizante pensada para dois jogadores em turnos, onde vence o jogador com mais pontos de um conjunto arbitrário de partidas.

Os jogadores 1 e 2 deverão escolher os seus papéis entre “Armador” e “Andarilho”. O armador deverá esconder os ovos podres em um terreno representado por uma matriz de 5 x 5. Podendo esconder até 3 armadilhas de ovos por linha da matriz e até um total de 15 armadilhas por todo o terreno. O andarilho deve tentar atravessar o terreno sem cair em nenhuma das armadilhas de ovos escondida.

Ao iniciar o game um menu deve ser exibido como o demonstrado abaixo:

Opções:

- 1 - Definir Armador
- 2 - Plantar Armadilhas
- 3 - Iniciar com Andarilho
- 4 - Mostrar o placar
- 0 - Finalizar o Jogo

A primeira coisa que o jogador que estiver utilizando o teclado deve fazer é definir o armador. Ao selecionar a opção 2 será impresso na tela “Qual jogador plantará as armadilhas? [1 ou 2]” e então o programa deve esperar até que o usuário insira um número válido (1 ou 2). O número informado determina qual jogador será o armador e o número não escolhido definirá o jogador que será o andarilho. Ao final essa configuração de jogadores deverá ser impressa na tela seguida pelo menu, como no exemplo abaixo:

O armador é o jogador: 1  
O andarilho é o jogador: 2

Agora o armador pode esconder as armadilhas de ovos podres no terreno. Ao selecionar a opção 2, o terreno sem armadilhas preenchido com letras ‘A’ será exibido na tela:

Observe o terreno abaixo:

```

0 1 2 3 4 5
1 A A A A A
2 A A A A A
3 A A A A A
4 A A A A A
5 A A A A A

```

A impressão do terreno é seguida da pergunta “Jogador [numero do jogador] , você pode esconder até 3 ovos podres por linha do terreno.”, onde o número do jogador faz referência ao jogador que foi configurado como armador. Em seguida o jogador é perguntado: “Em qual coluna da linha ,1 você quer esconder ovos podres? [1 a 5]”, o número da linha deve ser adicionado de 1 sempre que o armador informar que não deseja continuar colocando armadilhas nessa linha ou

após ter colocado três armadilhas na linha. Não deve ser possível plantar as armadilhas se o armador não tiver sido definido com a opção 1 do menu.

Depois que o armador esconder de 1 a 3 armadilhas em todas as linhas do terreno, o mapa com as armadilhas representadas pela letra “O” deve ser exibido:

Esse é o terreno com as armadilhas plantadas:

```
0 1 2 3 4 5
1 O A A O A
2 O O A A O
3 A O A A A
4 A O A A A
5 A O A A A
```

Depois das que as armadilhas estiverem plantadas, o jogador armador pode redefinir o terreno de armadilhas ou selecionar a opção 3 do menu, dando a vez para o jogador andarilho. Quando a opção 3 é selecionada, 100 linhas devem ser impressas. Na primeira linha nada deve ser impresso, na segunda linha o caractere ‘=’ deve ser impresso, na terceira linha dois caracteres ‘=’ devem ser impressos e assim por diante. Exemplo:

```
=
==
===
====
=====
=====
=====
=====
...
```

Então, o jogador andarilho receberá na tela duas mensagens:

São válidos os espaços: [1, 2, 3, 4, 5]  
Escolha sabiamente um dos espaços válidos

Os espaços válidos são calculados para cada linha do terreno. Para a primeira linha do terreno todos os espaços são válidos, para as demais linhas são válidos os espaços compreendidos entre o espaço escolhido na linha anterior -1 e o espaço escolhido pela linha anterior +1. Atenção: é importante verificar se o cálculo dos espaços válidos não extrapola os limites da matriz do terreno.

Quando um espaço válido é escolhido, o programa deve avaliar se o andarilho pisou ou não em uma armadilha de ovos podres. Então, caso o andarilho tenha escolhido um espaço no mapa preenchido com a letra “A”, é dado para ele a oportunidade de mudar de linha no terreno e escolher um outro espaço válido para tentar a sorte. Caso o andarilho tenha escolhido um espaço no mapa preenchido com a letra “O”, a seguinte mensagem deverá ser exibida: “Eca! Você pisou em um ovo podre e perdeu”, seguida do menu do jogo.

Sempre que o andarilho conseguir chegar na última linha do mapa sem pisar em uma armadilha, o jogador que o representa (1 ou 2) ganha um ponto e recebe a mensagem “Você atravessou o terreno sem cair em nenhuma armadilha! Parabéns!!!!”. Já para todas as vezes em que o andarilho pisar em uma armadilha, o jogador que representa o jogador armador ganha 1 ponto.

A partir do menu, a qualquer momento, os jogadores podem finalizar o jogo selecionando a opção 0 ou selecionar a opção 4 para ver quem tem mais pontos desde que o jogo foi iniciado. Como pode ser visto no formato abaixo:

Pontuação do jogador 1: 2  
Pontuação do jogador 2: 1

## Dicas do arquiteto de software

Um arquiteto de software já pode se dedicar sobre o problema e recomenda a você programador que:

1. Utilize um dicionário para armazenar as configurações do jogo como: a quantidade máxima de armadilhas, o número do jogador armador, o número do jogador andarilho, a pontuação o jogador 1 e a pontuação do jogador 2;
2. Construa um procedimento para redefinir o terreno para a sua configuração inicial;
3. Escreva uma função que verifique se as entradas dadas pelos usuários são válidas. Essa função deve retornar valores inteiros quando forem válidos ou requerer um novo valor até que uma entrada válida seja apresentada. Todas as entradas dos programa devem ser validadas e não será permitido ao usuário prosseguir enquanto uma entrada válida não for apresentada;
4. Escrever um procedimento para imprimir o terreno;
5. Escrever um procedimento para que o armador plante as armadilhas;
6. Escrever um procedimento que possibilite o caminhar do andarilho pelo mapa, verificando quando ele cai em armadilhas ou não;
7. Escrever uma função que retorne uma lista de espaços válidos para serem informados ao andarilho;

Apesar de o arquiteto entender que essas são boas opções de organização do código, você enquanto programador está livre para definir outras funções, procedimentos ou estruturas de dados que achar conveniente.