

Introdução à Programação - 2025-1 - Prova de Suficiência

A prova de suficiência abrange todo o conteúdo da disciplina (6 créditos). O conteúdo é o que consta no Plano de Ensino da disciplina (disponível no AVA3). As únicas estruturas de dados permitidas para a solução de problemas são strings e vetores. Nenhuma outra estrutura ou biblioteca é permitida, a não ser o uso do `"import java.util.Scanner;"` para fazer uso de `"Scanner teclado = new Scanner(System.in);"` e fazer leituras do teclado. E também o `import "import java.util.Random;"` para ser utilizado na rotina que precisa randomizar um número.

A prova de suficiência da disciplina de Introdução à Programação deve ser cumprida individualmente.

Prazo máximo para entrega: 27/02/2025 às 23h59min (quinta)

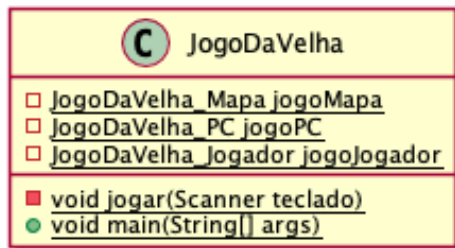
Definições

Com base nas seguintes descrições, implemente o problema do: **Jogo da Velha**.

ATENÇÃO: deve seguir fielmente a descrição a baixo.

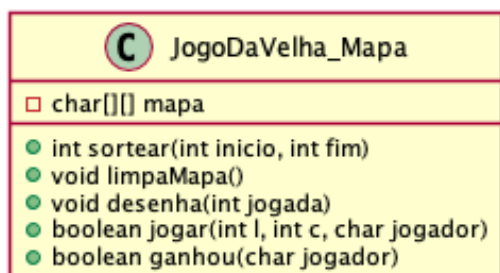
Ter quatro classes descritas a seguir ...

Classe JogoDaVelha



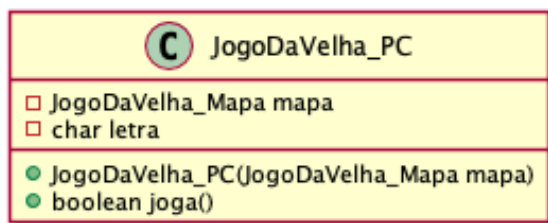
- o método main declarar a única ocorrência do objeto teclado da classe Scanner para permitir ler dados do console usando - o teclado;
- o método main instância os objetos das classes: JogoDaVelha_Mapa, JogoDaVelha_PC e JogoDaVelha_Jogador, para iniciar os - seus atributos;
- o método main controla se “Deseja jogar novamente”;
- o método jogar “limpa” o mapa a cada nova partida;
- o método jogar sorteia quem inicia a partida (jogador ou PC);
- o método jogar tem a variável que conta quantas jogadas já foram executadas;
- o método jogar chama o método de desenhar o mapa;
- o método jogar tem o laço que controla as jogadas, alternando a vez entre jogador e PC.

Classe JogoDaVelha_Mapa



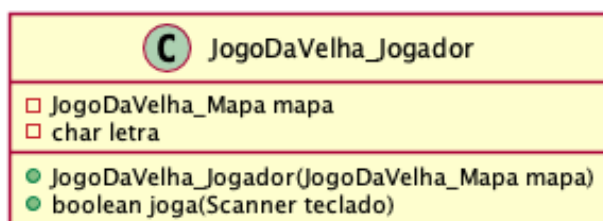
- o mapa é definido pelo tipo primitivo char;
- o método sortear pode ser buscado na internet;
- usar o caráter ' ' (espaço em branco) para limpar o mapa;
- a cada jogada testar quem venceu a partida (jogador ou PC);
- considerar o caso de empate no final da partida;
- testar no método jogar se a linha e coluna recebida representa uma célula livre no mapa, e em caso positivo marcar nesta célula quem jogou (usar a letra).

Classe JogoDaVelha_PC



- a jogada do PC deve ser representada pelo caráter “O” (sua letra);
- usar o método sortear para definir a próxima jogada do PC;
- a cada jogada testar se o PC venceu, e neste caso imprimir " ... PC GANHOU!".

Classe JogoDaVelha_Jogador

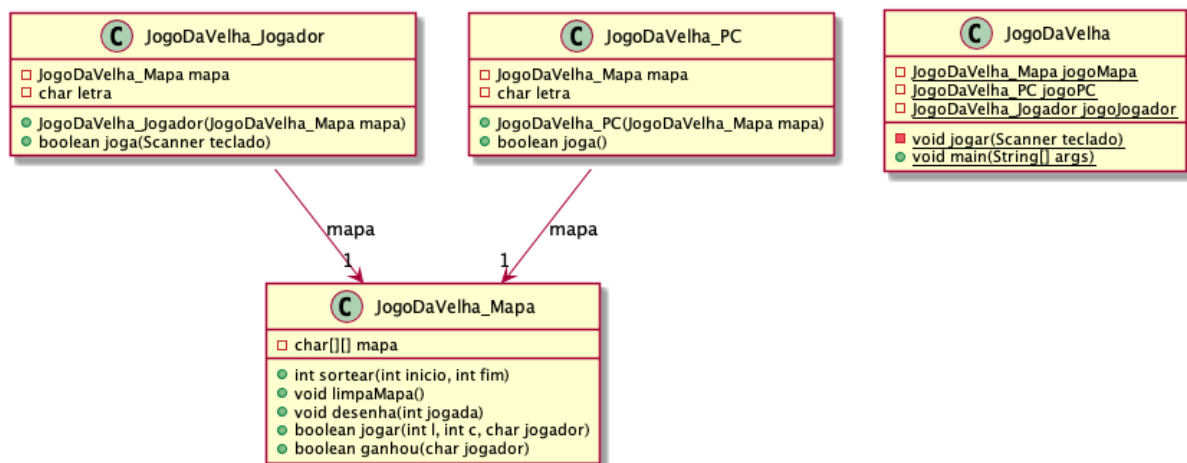


- a jogada do jogador deve ser representada pelo caráter “X” (sua letra);

- usar o método joga para ler linha e coluna do teclado;
- a cada jogada testar se o jogador venceu, e neste caso imprimir " ... Jogador GANHOU!";
- verificar se os valores de linha e coluna são válidos (dentro dos limites do mapa).

Atenção

O código **deve** seguir as informações descritas nos diagramas de Classes referente aos nomes das classes, atributos, métodos, parâmetros e qualificadores (visibilidade: public/private). Novas variáveis primitivas podem ser criadas dentro dos métodos, mas não podem ser criadas novas classes, atributos e métodos (só os que foram definidos no diagrama de classes abaixo).



| ícone atributo | ícone método | visibilidade |
|----------------|--------------|--------------|
| □ | ■ | private |
| ○ | ● | public |

Exemplo da execução

As mensagens mostradas ao executar o código deste trabalho devem seguir o que é mostrado abaixo.

```
----- .. jogada: 0
|   |   |   |
-----
|   |   |   |
-----
|   |   |   |
-----

Jogador ..
  linha:
1
  coluna:
1
----- .. jogada: 1
|   |   |   |
-----
|   | x |   |
-----
|   |   |   |
-----

PC[1,0]
----- .. jogada: 2
|   |   |   |
-----
| o | x |   |
-----
|   |   |   |
-----

Jogador ..
  linha:
0
  coluna:
0
----- .. jogada: 3
| x |   |   |
```

```
-----
| o | x |   |
-----
|   |   |   |
-----
PC[0,1]
----- .. jogada: 4
| x | o |   |
-----
| o | x |   |
-----
|   |   |   |
-----
Jogador ..
  linha:
2
  coluna:
0
----- .. jogada: 5
| x | o |   |
-----
| o | x |   |
-----
| x |   |   |
-----
PC[0,2]
----- .. jogada: 6
| x | o | o |
-----
| o | x |   |
-----
| x |   |   |
-----
Jogador ..
```

```
linha:
2
coluna:
1
----- .. jogada: 7
| x | o | o |
-----
| o | x |   |
-----
| x | x |   |
-----
PC[2,2]
----- .. jogada: 8
| x | o | o |
-----
| o | x |   |
-----
| x | x | o |
-----
Jogador ..
linha:
1
coluna:
2
----- .. jogada: 9
| x | o | o |
-----
| o | x | x |
-----
| x | x | o |
-----
----- .. jogada: 9
| x | o | o |
-----
```

```
| O | X | X |
```

```
-----
```

```
| X | X | O |
```

```
-----
```

```
... EMPATOU!
```

```
Deseja jogar novamente (s/n)?
```

```
s
```

```
----- .. jogada: 0
```

```
|   |   |   |
```

```
-----
```

```
|   |   |   |
```

```
-----
```

```
|   |   |   |
```

```
-----
```

```
PC[2,1]
```

```
----- .. jogada: 1
```

```
|   |   |   |
```

```
-----
```

```
|   |   |   |
```

```
-----
```

```
|   | O |   |
```

```
-----
```

```
Jogador ..
```

```
  linha:
```

```
1
```

```
  coluna:
```

```
1
```

```
----- .. jogada: 2
```

```
|   |   |   |
```

```
-----
```

```
|   | X |   |
```

```
-----
```



```
|   | o |   |
-----
PC[2,0]
----- .. jogada: 3
|   |   |   |
-----
|   | x |   |
-----
| o | o |   |
-----
Jogador ..
  linha:
1
  coluna:
0
----- .. jogada: 4
|   |   |   |
-----
| x | x |   |
-----
| o | o |   |
-----
PC[0,2]
----- .. jogada: 5
|   |   | o |
-----
| x | x |   |
-----
| o | o |   |
-----
Jogador ..
  linha:
0
  coluna:
```

```

0
----- .. jogada: 6
| x |   | o |
-----
| x | x |   |
-----
| o | o |   |
-----

PC[0,1]
----- .. jogada: 7
| x | o | o |
-----
| x | x |   |
-----
| o | o |   |
-----

Jogador ..
  linha:
1
  coluna:
2
... Jogador GANHOU!
----- .. jogada: 8
| x | o | o |
-----
| x | x | x |
-----
| o | o |   |
-----

-----
Deseja jogar novamente (s/n)?
s
----- .. jogada: 0

```

```
|   |   |   |
-----
|   |   |   |
-----
|   |   |   |
-----
PC[2,2]
----- .. jogada: 1
|   |   |   |
-----
|   |   |   |
-----
|   |   | O |
-----
Jogador ..
  linha:
1
  coluna:
1
----- .. jogada: 2
|   |   |   |
-----
|   | X |   |
-----
|   |   | O |
-----
PC[1,2]
----- .. jogada: 3
|   |   |   |
-----
|   | X | O |
-----
|   |   | O |
-----
```

```

Jogador ..
  linha:
2
  coluna:
0
----- .. jogada: 4
|   |   |   |
-----
|   | x | o |
-----
| x |   | o |
-----
PC[0,2]
... PC GANHOU!
----- .. jogada: 5
|   |   | o |
-----
|   | x | o |
-----
| x |   | o |
-----

-----
Deseja jogar novamente (s/n)?
n

```

--- FIM ---