

MAC115 - Introdução à Computação IF

Terceiro Exercício Programa - Primeiro Semestre de 2006

Entregar até: 30/06/2006

1 Introdução

Neste Exercício Programa cada dupla deve implementar uma versão funcional do jogo de Trilha. As instruções sobre como o jogo funciona podem ser encontradas em: <http://www.angelfire.com/ab/jogos/Tradicionais/trilha.html>.

2 O Exercício

Neste exercício você deve fazer um programa em Java que permita que dois usuários joguem o jogo da Trilha. O programa deve:

- controlar as fases do jogo. A fase inicial consiste na colocação das peças, nove por usuário. Na segunda fase é possível apenas movimentar peças a posições adjacentes. Finalmente, a terceira fase começa quando um dos jogadores possui apenas 3 peças, e neste caso este não precisa mais respeitar a regra de se movimentar apenas a posições adjacentes.
- verificar se as jogadas são válidas;
- verificar a formação de trilhas;
- detectar se algum dos jogadores ganhou o jogo.

A interface com o usuário pode ser feita através do teclado, e o tabuleiro pode ser desenhado apenas com caracteres de texto, como abaixo:

```
X-----0      fase 0
| .---0---. |
| | .-X-. | |   vez do jogador 0
.---.-. .-.-X
| | .-0-. | |
| .---.---X |
.-----.
```

Para uniformizar a forma que os jogadores devem interagir com o programa usaremos o seguinte sistema de coordenadas (x, y) :

- onde x corresponde ao círculo, sendo 0 o mais externo, e dois o mais interno;
- onde y corresponde à posição no círculo (sendo 0 a posição correspondente ao 12 no relógio e a contagem horária);

- no exemplo acima temos as posições (0, 2), (0, 7), (1, 3) e (2, 0) com X.

Para este jogo, você deve implementar uma classe `Trilha.java` que simula o jogo e deve ter os seguintes métodos:

- `void novoJogo()`: Limpa o tabuleiro para um novo jogo;
- `void imprimeTabuleiro()`: imprime o tabuleiro atual;
- `int [][] devolveMatriz()`: devolve uma matriz 3x7 com o atual estado do tabuleiro;
- `devolveFaseAtual()`: devolve 0, 1 ou 2 conforme a fase atual ou -1 se acabou o jogo;
- `boolean efetuaJogada(int ci1, int ci2, int cf1, int cf2)`: que devolve `true` e efetua a jogada caso ela seja válida. Caso contrário devolve `false`, sem alterar o tabuleiro. Para a primeira fase e para a retirada de peças este método só considera as duas primeiras coordenadas¹;
- `boolean formouTrilha(int c1, int c2)`: verifica se a colocação da peça nas coordenadas dada formou uma trilha;
- Lembre-se que você pode criar métodos adicionais como: `int mobilidade(int jogador)` - para verificar se um jogador está bloqueado.

Além desta classe você deve criar uma classe `Principal.java` com um método `main` para controlar e imprimir uma partida.

As regras dos EPs anteriores continuam valendo.

¹Note que é bem interessante decompor este método em vários sub-métodos