

Programação Orientada por Objetos

Primeiro trabalho prático
Semestre de Inverno 2018/2019

Objetivos: Neste trabalho pretende-se que os alunos adquiram prática de utilização de herança e polimorfismo usando a abordagem MVC na implementação de um jogo em consola de texto.

Entrega: Cada grupo entregará no *moodle* da sua turma, até dia 5 de novembro, os ficheiros fonte (java) do projeto realizado e o documento que descreve a implementação do trabalho contendo o diagrama de classes.

Enunciado: Implemente o jogo designado por *Snake* que permite escolher a direção de deslocamento de uma cobra, cuja cabeça é representada por @ e cada secção por #, tentando comer as 10 maçãs (representadas por O) sem colidir com os obstáculos (representados por X). A cobra desloca-se numa arena em que o limite superior é adjacente ao inferior e o esquerdo é adjacente ao direito. A figura 1 apresenta o aspeto de um estado possível deste jogo.

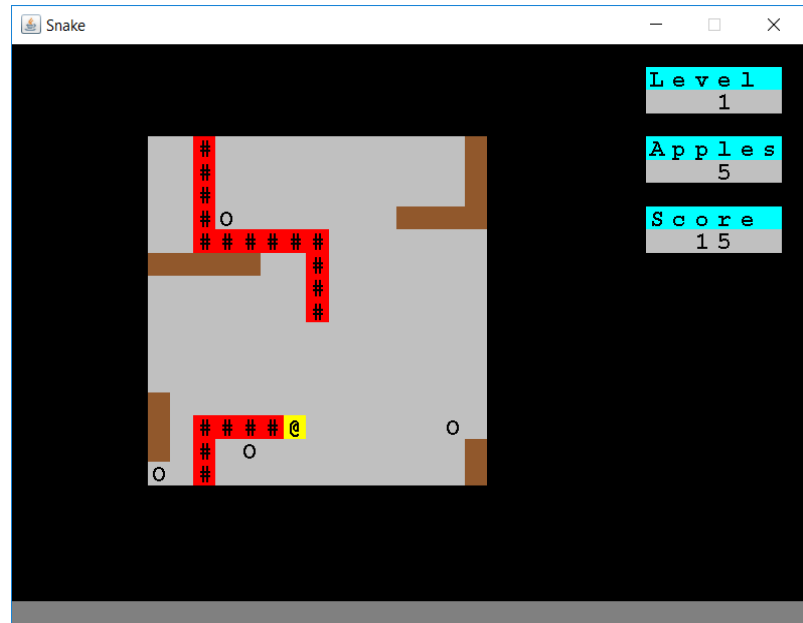


Figura 1: Aspeto do jogo

Os níveis são descritos no ficheiro de texto (levels.txt) em que a primeira linha de cada nível indica o número do nível e as dimensões da arena (linhas x colunas), e as restantes descrevem o estado inicial da arena em que cada letra corresponde a um tipo de célula (@-Cabeça da cobra; A-Maçã; X-Obstáculo).

As maçãs descritas no ficheiro são apenas as iniciais. Quando uma é comida aparece outra (até ao limite de 10) em lugar aleatório, a cobra cresce mais 4 secções e são atribuídos 4 pontos.

As secções da cobra são adicionadas à cauda à medida que ela se desloca. Inicialmente a cobra tem 4 secções e desloca-se para cima. A cada 10 movimentos, a cobra perde uma secção e é retirado 1 ponto.

A figura 2 apresenta o troço do ficheiro que descreve a arena inicial da figura 1.

```
#1 15 x 15
                X
                X
                X
                XXXX
XXXXX
A              A
              @
X              X
X              X
X              X
A              X
```

Figura 2: Troço de levels.txt

O programa deve construir cada nível a partir da descrição do ficheiro e deslocar a cobra a cada meio segundo no sentido da última tecla de direção premida.

Pode ser experimentada uma versão do jogo pretendido em “[Snake.zip](#)”.

Este *zip* contém um *jar* com o programa de demonstração, o ficheiro de descrição dos níveis, o *jar* da biblioteca ConsolePG e uma pasta “src” com os ficheiros fonte de algumas das classes já implementadas. Destas classes também é possível depreender algumas das classes que são necessárias definir e alguns dos métodos públicos.

A figura 3 mostra as classes principais do programa, apresentando em fundo azul as totalmente implementadas. Em fundo amarelo estão representadas as hierarquias de classes que devem ser definidas para cada tipo de célula no modelo e na visualização.

Implemente o jogo descrito seguindo a abordagem MVC.

1. Defina uma hierarquia de classes que represente os vários tipos de células (classe *Cell* e suas derivadas) e as classes principais do **modelo** (classes *Game* e *Level*) que implementam a lógica do jogo (as regras). O código destas classes será reutilizado na implementação deste jogo em *Android* sem necessidade de alterações.

2. A componente de **controlo** que interage com o utilizador (classe *Snake*), processa as teclas. O **controlo** recebe notificações e chama métodos do **modelo** para evoluir no estado do jogo e métodos da componente de **visualização** para redesenhar as células.
3. A **visualização** usa a classe *Console*, da biblioteca ConsolePG incluída no zip, e as classes do package isel.poo.console, faltando definir as classes *StatusPanel*, *CellTile* e derivadas que implementam a apresentação de cada tipo de célula.

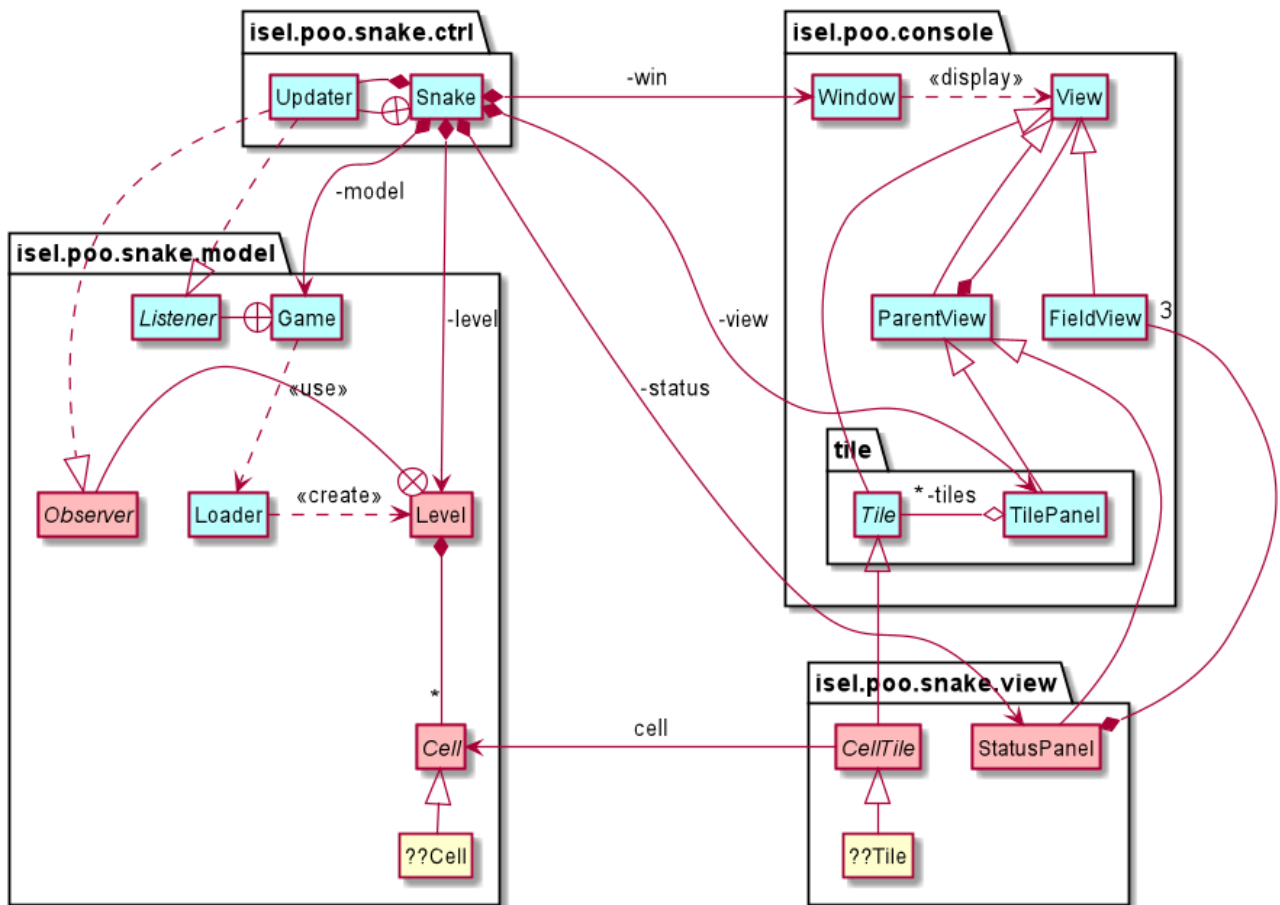


Figura 3: Diagrama de classes

Opcionalmente poderão ser implementados outros tipos de elementos:

- Ratos que se deslocam, em cada 4 movimentos da cobra, para uma posição adjacente aleatória. A cobra poderá comer o rato, crescendo 10 secções e ganhando 10 pontos.
- Outras cobras que se deslocam automaticamente mudando de direção quando encontram um obstáculo e também podem comer maçãs e ratos. Estas cobras só morrem quando ficam encurraladas, começando a encurtar pela cauda. A cobra do jogador pode comer a cabeça destas cobras depois de mortas, sendo o número de secções e a pontuação igual a 10 mais 2 vezes o comprimento atual da cobra comida.

O programa de demonstração fornecido, já implementa estes elementos opcionais.