Universidade de Évora

Programação II

Trabalho Prático - Jogo da Vida

Eunice Devesse - 50710

Leonel Caetano - 53208

Janeiro de 2024

Contents

Chapter 1

Introdução

O presente trabalho tem como objectivo realizar a simulação duma versão do jogo da vida, esta com coelhos e cenouras que habitam um prado verde. Neste jogo, o prado é descrito pelo seu tamanho e pela posição inicial dos coelhos e das cenouras. Este prado também possui um parâmetro starveTime que especifica o número de unidades de tempo de simulação que um coelho sobrevive sem ser alimentado.

Nesta simulação, as entidades (coelhos e cenouras) interagem conforme regras específicas a cada unidade de tempo.

1.1 Regras para implementação do jogo

1. Se uma célula contém um coelho e pelo menos um de seus vizinhos é uma cenoura, o coelho se alimenta da cenoura durante o período de tempo e permanece na mesma célula ao final desse período.

2. Se uma célula contém um coelho e nenhum de seus vizinhos é uma cenoura, o coelho passa fome durante o período de tempo.

3. Se um coelho passa fome por um número específico de períodos de tempo ("starveTime"), ele morre e desaparece da célula.

4. Se uma célula contém uma cenoura e todos os seus vizinhos são cenouras ou células vazias, a cenoura permanece na célula ao final da unidade de tempo.

5. Se uma célula contém uma cenoura e pelo menos um de seus vizinhos é um coelho, a cenoura é comida pelo coelho e desaparece.

6. Se uma célula contém uma cenoura e dois ou mais dos seus vizinhos são coelhos, um coelho recém-nascido ocupa a célula ao final da unidade de tempo.

7. Coelhos recém-nascidos estão bem alimentados, significando que podem aguentar "starveTime" unidades de tempo sem se alimentar.

8. Se uma célula está vazia e menos de dois vizinhos são cenouras, ela permanece vazia.

9. Se uma célula está vazia e pelo menos dois dos seus vizinhos são cenouras, e no máximo um dos vizinhos é um coelho, uma nova cenoura nasce na célula.

10. Se uma célula está vazia e pelo menos dois dos seus vizinhos são cenouras e pelo menos dois dos seus vizinhos são coelhos, um novo coelho nasce na célula.

Chapter 2

Implementação em Java

Para a implementação deste projecto, foram criadas 5 classes, nomeadamente:

- Grassland;
- Simulation;
- Rabbit;
- Carrots;
- Piece;

2.1 Grassland.java

A classe Grassland conta com a definição de 3 métodos adicionais. O método **initialize()** inicializa todas as células do prado com o valor EMPTY.

O método **copyMeadow(Piece meadow[][], Piece meadow2[][])** copia o meadow para o objecto meadow2 sombra que conterá os movimentos atuais no prado.

O método **search_execute(int type, int x, int y, Piece positionCurrent)**

A função **timestep()**

2.2 Simulation.java

A classe Simulaton contém os Gráficos para a representação do prado. Esta classe já foi definida para apresentar a interface da simulação do jogo. Nela, se encontram métodos responsáveis pelo posicionamento aleatório das peças (cenouras e coelhos), bem como as caracteristicas visuais para cada entidade. Na implementação do código, foi alterada a funçao random, de modo que as cenouras e os coelhos fossem dispersos de maneira aleatória por todo o prado.

2.3 Rabbit.java e Carrot.java

As classes rabbit e Carrot, herdam os atributos e os métodos da classe Piece. Esta classe possui um contrutor que define

2.4 Piece.java

Esta classe implementa o método **toReproduce(int type, int posX, int posY, Piece objectPiece[][])** que adiciona um objecto do tipo cenoura ou coelho na posiçao do Meadow, passada como argumento. Também foi implementado o método **toKill(int posX, int posY, Piece objectPiece[][])** que elimina as cenouras ou os coelhos de acordo com as regras.