

SCC0604 - Programação Orientada a Objetos

Manual do Sistema

Pacman

Nome: Lucas Alves Roris N°USP: 11913771

O projeto todo é dividido em 3 pacotes: Engine, Elementos do sistema e Interface Gráfica. Cada uma será desmembrada e explicada abaixo:

1. Interface Gráfica

A interface gráfica possui apenas uma classe, a TelaJogo, que realiza todas as funções necessárias para fazer o jogo tomar forma. No seu método start() é criado todas as variáveis necessárias para abrir uma aplicação windows e a classe da Engine, que realiza as funções vitais que fazem o jogo rodar.

Logo após o fim do método start, ela inicia o método game_loop(), que fica em um loop até o final do jogo. Basicamente o que o método faz é pegar os elementos que a engine modifica e desenha na tela, junto com informações como pontuação, vida e nível.

Ao chegar em um game over, a tela muda para uma tela de game over e o programa fecha após 3 segundos.

2. Engine

A engine possui 4 classes:

Engine:

A classe Engine representa a progressão do jogo. É ela que possui as regras de negócio do jogo, conecta as classes de mapa, sons, personagens, elementos, etc. Nela tem diversas funções auxiliares, mas a principal é o método processaJogo(), que é executado continuamente graças a interface gráfica, e nela possui o passo a passo do jogo.

Primeiro ela faz uma verificação de condições atuais do jogo, como número de vidas restantes, pac-dots restantes, etc. Depois verifica se os personagens já podem andar através de um timer individual dos fantasmas. Quando o timer ultrapassa o tempo que o fantasma pode ficar parado, ela faz o personagem andar e

reseta o contador. Após a verificação dos fantasmas, o jogo verifica se alguém teve colisão com algum elemento do jogo, caso o pacman tenha colidido com pac-dots ou frutas, ele ganha a respectiva pontuação, já caso ele acerte um fantasma, ele verifica se o fantasma está azul ou não, se sim, ele mata o fantasma e ganha a pontuação, se não, ele perde uma vida e reseta a posição dos personagens.

Caso tenha coletado todos os elementos do tipo pac-dot e power-pill, ele passa de nível, resetando os elementos e aumentando a dificuldade. E caso as vidas cheguem a zero, o jogo acaba.

Node:

A classe Node representa uma posição ocupável do mapa, e possui informações como o que a posição possui, as coordenadas para desenhar no mapa e quais vizinhos os fantasmas têm acesso e quais vizinhos o pac-man tem acesso.

Mapa:

A classe Mapa representa o layout do mapa em forma de um ArrayList de characters que os transforma em um ArrayList de Nodes, e tem a função de determinar quais as posições ocupadas, quais as direções possíveis de andar, etc.

Som:

A classe Som representa todos os sons do jogo. Ao iniciar ela, ela inicializa uma variável para cada som e fica esperando para quando alguma função a chamar, executa os sons sem ficar abrindo o arquivo .wav toda hora.

3. Elementos de sistema

O pacote de elementos do sistema possui todos os elementos que podem ser colocados no mapa. A principal classe é a classe Elemento, que deriva para todas as outras. Ela possui o estado atual do elemento, a posição dele, a imagem que o representa, e alguns métodos genéricos.

A outra classe importante é a Personagem, que representa um elemento que pode andar no mapa, que é derivado para Fantasma e Pacman.

Fantasma:

A classe fantasma representa o inimigo do pac-man, e possui variáveis de status, nome e qual a posição inicial dele. Ele anda conforme uma variável chamada pathToFlow, que determina para onde ele deve ir. Esse caminho é determinado por uma posição aleatória no mapa, e para descobrir como chegar lá, é utilizado o conceito de busca em largura (BFS) a partir do mapa.

Basicamente o BFS começa da posição atual do fantasma e, olhando as posições possíveis de deslocamento pelo mapa, vai procurando pela posição aleatória determinada previamente. Quando achar, ela para o processo e refaz o caminho, armazenando quais posições deve seguir para alcançar a posição sorteada.

Cada vez que o fantasma atinge seu objetivo, é feito um novo BFS para uma posição aleatória nova.

A imagem desenhada no mapa depende de qual direção o personagem está andando.

Fantasma Inteligente:

Essa é uma classe derivada da Fantasma, porém, ao invés de fazer a busca em largura por uma posição randômica, a busca será feita através da posição atual do pac-man. E como a posição do pac-man muda toda hora, a cada passo dado, o BFS recalcula o caminho para saber se o caminho seguido é o mesmo ou não.

Porém, quando um Fantasma Inteligente fica azul, ele deixa de perseguir o pac-man e passa a se comportar igual um fantasma normal, com as posições aleatórias. Isso se dá ao fato de que seria muito fácil o pac-man pegar a power-pill e esperar que os fantasmas inteligentes se matassem.

Pac-Man:

Essa é a classe que controla o personagem principal. Basicamente ela possui o método de andar relacionado com uma variável chamada direção, e essa variável é modificada através do direcional do teclado, por um evento da interface gráfica.

A imagem desenhada no mapa depende de qual direção o personagem está andando.

Outras classes desse pacote são os objetos que valem pontuação e geram efeitos no jogo, como o Pac-Dot, Power-Pill e Fruta.

Pac-Dot:

A classe que representa a menor unidade do jogo, valendo 10 pontos e possuindo 194 por nível.

Power-Pill:

A classe que representa uma pílula que, ao consumida, deixa todos os fantasmas azuis por 10 segundos. O efeito dos fantasmas azuis permite que o pac-man coma eles e ganhe 200 pontos, dobrando a cada fantasma morto, no

máximo de 1600 pontos. A cada nível a pílula dura 1 segundo a menos, até um mínimo de 2 segundos.

Fruta:

A classe que representa uma fruta. No nível 1 possui duas cerejas que valem 100 pontos. No nível 2 possui dois morangos que valem 300 pontos. Por fim, no nível 3 possui duas laranjas que valem 500 pontos. Após o nível 3, as frutas se repetem, mas a pontuação continua aumentando.

As frutas aparecem quando o pac-man consumiu 58 e 135 pac-dots todos os níveis.