

Relatório MC322 - Pacman

Nomes:

Régis Gabetta de Souza

Gabriel Da Silva Costa

Lucas de Oliveira Silva

RAs:

223965

216158

220715

Funcionamento do programa:

Ao começar o jogo, será perguntado ao usuário via console se ele quer partir de um mapa aleatório ou de um mapa padrão. Após escolher o tipo de mapa, o jogo começará!

O Pacman nascerá no canto superior esquerdo da tela inicialmente na direção direita. Ao mesmo tempo, os 4 fantasmas nascerão no spawn e começarão a perseguir o Pacman. Note que o spawn dos fantasmas é gerado aleatoriamente, sendo uma região quadrada sem pastilhas. Caso o Pacman coma um power up, ele tem 5 segundos para caçar os fantasmas, ganhando 100 pontos ao comer o power up e 500 pontos para cada fantasma caçado.

O Pacman tem 3 vidas, e caso consiga comer todas as pastilhas, ele passa para o próximo nível. Caso o Pacman colida com algum fantasma sem ter o power up, ele perderá uma vida. Nesse caso, ambos o Pacman e os Fantasmas serão direcionados aos seus respectivos pontos de respawn. Quando o pacman atinge 10.000 pontos ele ganha uma vida extra.

Código:

Nosso Pacman é um projeto visual. Para tratar a interface do usuário e interação com o mesmo, estamos usando 4 classes do Java: JFrame, JPanel, ActionListener e KeyAdapter. Escolhemos usar o JFrame, e JPanel para Criar a janela gráfica e o visual do jogo. Para processar os eventos que acontecem no jogo, usamos o ActionListener. O KeyAdapter é responsável pela leitura e interpretação das teclas clicadas pelo usuário.

As classes internas do jogo se dividem principalmente em dois grupos, objetos do jogo (Game Objects) e controladores de tais objetos (Controllers). Dentro dos controllers, temos o BoardController, que é responsável por controlar os obstáculos e o visual do mapa. Também temos a classe KeyController, que é uma extensão do KeyAdapter, controlando as teclas digitadas pelo usuário. Também temos o ObstacleController, uma classe para controlar os obstáculos do jogo. O GameController é a classe principal, que será responsável por controlar o fluxo do jogo a partir tanto dos outros controllers como também dos game objects.

Game Objects: são objetos do jogo. Neles se encontram o Pacman, os Ghosts, o PowerUp, as pastilhas coletáveis (Circle), e paredes.

Também é interessante destacar que temos um subdiretório chamado utilities, que são classes utilizadas para fazer funções específicas que os Controllers ou os Game Objects poderão utilizar. Um bom exemplo é a classe MapGenerator, que tem como função gerar o mapa que será controlado pelo BoardController.

Geração de Mapas:

Para gerar o mapa, o programa começa criando uma matriz e sorteando uma posição. Logo após ela é preenchida com números representando as paredes do mapa, onde todas têm formato de “L” (detalhes do preenchimento

<https://www.geeksforgeeks.org/tiling-problem-using-divide-and-conquer-algorithm/>). Com essa matriz inicial é gerado uma nova com dimensões dobrada onde “1” significa parede, “0” espaço vazio, “2” power-ups e “-1” spawn area. Depois de gerada essa representação do mapa é criado um objeto obstáculo para cada elemento do jogo.

