

Meta 1

Programação Avançada

Daniel Bravo – 2021137795

Índice

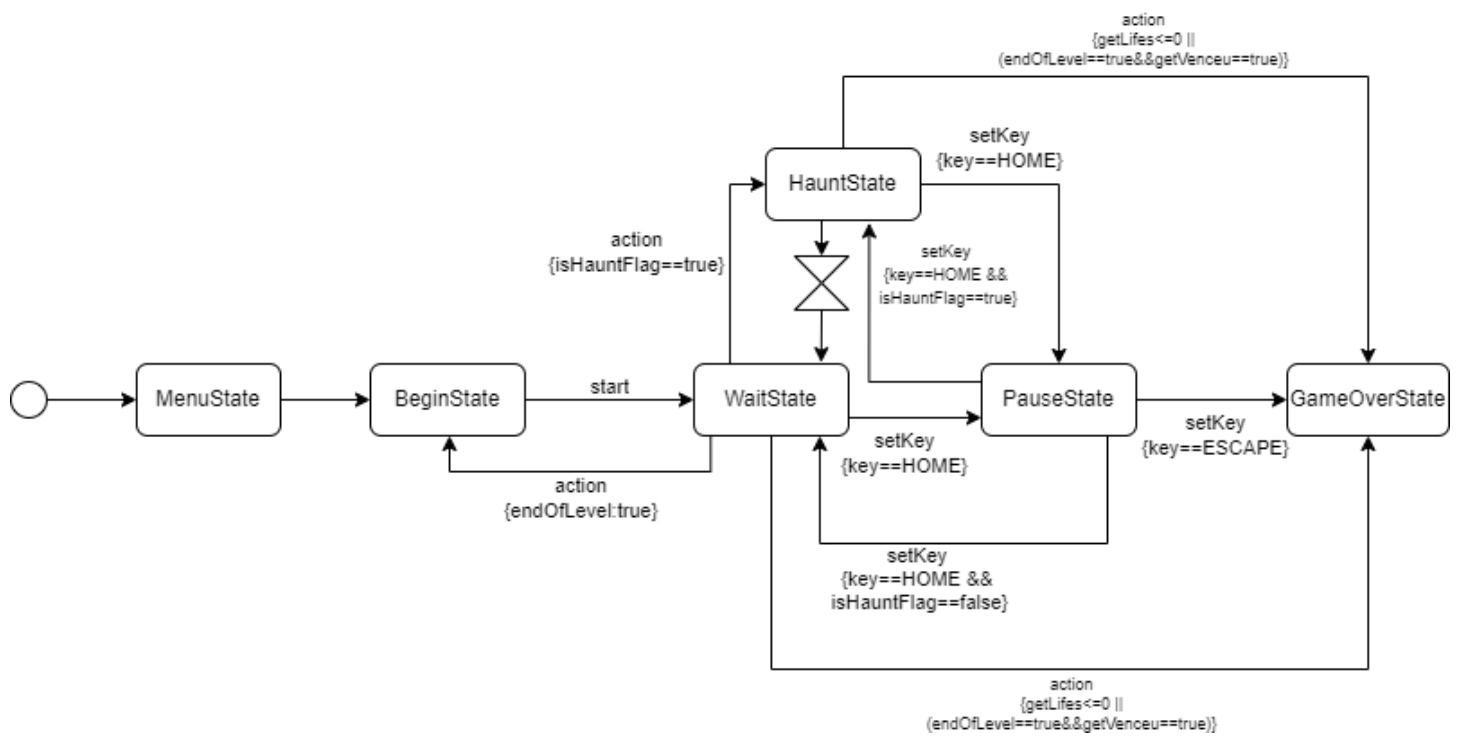
Descrição das opções e decisões tomadas	2
Diagrama de estados	3
Descrição das classes	4
Descrição do relacionamento entre as classes	6
Funcionalidades implementadas	7

Descrição das opções e decisões tomadas

Este trabalho consiste na implementação do jogo Pacman em Java, sendo nesta meta usado uma interface em texto.

Na minha implementação os objetos estão á parte do maze pois desta forma é permitido que esses mesmos objetos estejam numa mesma posição que uma moeda, fruta, etc sem que haja a necessidade de criar variáveis para evitar o desaparecimento destas.

Diagrama de Estados



Descrição das Classes

Blinky, Clyde, Inky, Pinky

Representam respetivamente os diferentes tipos de fantasma presentes no jogo e guardam o seu caracter representativo, assim como as suas coordenadas

GameBWData

Classe que possui toda a lógica do jogo e que é dona de todos os objetos.

IGhost

Base das classes fantasma

MazeElement

Classe que implementa a classe IMazeElement de forma a permitir o armazenamento de um char

Pacman

Representa o pacman e guarda o seu caracter representativo, assim como as suas coordenadas

IGameBWState

Classe interface dos estados

GameBWStateAdapter

Classe que implementa *IGameBWState* que é depois extendida pelos outros estados

GameBWState

Enum que contem o nome dos diferentes estados e responsável por os criar

Direction

Enum que contem as diferentes direções possíveis

BeginState

Classe que representa o estado responsável por iniciar um jogo ou um novo nível

GameOverState

Classe que representa o estado responsável por terminar o jogo

HauntState

Classe que representa o estado no qual após o pacman comer a fruta os fantasmas ficam vulneráveis

MenuState

Classe que representa o estado responsável pelo menu inicial

PauseState

Classe que representa o estado no qual o jogo se encontra em pausa

WaitState

Classe que representa o estado no qual o jogo está em ação, mas os fantasmas não se encontram vulneráveis

Descrição do relacionamento das classes

A interface *IGameBWState* é implementada pela classe *gameBWStateAdapter*, que é extendida pelas classes que representam estados da máquina de estados: *BeginState*, *GameOverState*, *HauntState*, *MenuState*, *PauseState*, *WaitState*.

Estes estados são criados pela classe *GameBWState* e geridos pela classe *GameBWStateAdapter*.

A classe *GameBWContext* tem acesso a estes estados através da interface *IGameBWState*. Para além de ter acesso a estes estados também tem acesso à *GameBWData* que contém toda a lógica do jogo assim como é dona das classes *Pacman*, *Blinky*, *Pinky*, *Clyde* e *Inky*, onde estas quatro últimas possuem todas a mesma interface.

Por último temos a classe *GameBWUI* que interage diretamente com o utilizador e que transmite os inputs do utilizador para o jogo através da classe *GameBWContext*, assim como permite mostrar ao utilizador o que está a acontecer dentro do jogo.

Funcionalidades Implementadas

- Ecrã inicial
- Controle do personagem Pac-Man com teclado
- 4 fantasmas com comportamentos diferentes
- Vários labirintos com níveis de dificuldade crescente obtidos através da leitura de ficheiros
- Zona Warp para teletransporte do Pac-Man
- Bolas para recolher e ganhar pontos
- Frutas que aparecem a cada 20 bolas recolhidas para ganhar pontos extras
- Bolas comestíveis que tornam os fantasmas vulneráveis e dão pontos
- Efeito de vulnerabilidade dos fantasmas e velocidade vai aumentando conforme o nível
- Opção de pausar jogo
- Aviso de Vitória

Funcionalidades não implementadas

- N/A