



UNIVERSIDADE ESTADUAL DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA

CONSTRUÇÃO DO JOGO PACMAN A PARTIR DO ALGORITMO DE DIJKSTRA

MANAUS, AM

2024

UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA

ALEXANDRE BRUNO MOTA DOS SANTOS

BEATRIZ DAMASCENO FERNANDES

MATEUS BASTOS MAGALHÃES MAR

MONIKE FREITAS DE SOUSA

SABRINA MILLANE OLIVEIRA DA SILVA

PROFESSOR WALTER CHARLES SOUSA SEIFFERT SIMOES

DISCIPLINA: ALGORITMOS E ESTRUTURA DE DADOS II

MANAUS, AM

2024

Sumário

Sumário.....	3
Descrição do problema.....	4
Solução proposta.....	5
Diagrama.....	6

Descrição do problema

Este trabalho tem como objetivo descrever o processo de desenvolvimento de um jogo de Pac-Man, um clássico dos videogames, utilizando a teoria dos grafos. O Pac-Man é um dos jogos mais populares da história, lançado em 1980 pela empresa japonesa Namco, em que, um personagem que se move por um labirinto, coletando pacdots enquanto evita fantasmas. A abordagem com grafos permitirá a modelagem eficiente do labirinto, fornecendo uma estrutura que facilita a navegação do Pac-Man e a implementação de estratégias para o movimento dos fantasmas.

O problema central consiste em criar um ambiente virtual que representa o labirinto no qual o Pac-Man se move. Este labirinto é modelado como um grafo, no qual cada célula é um nó e as conexões entre as células são arestas. O Pac-Man deve ser capaz de se mover pelos nós, coletando pacdots representadas por pontos específicos no grafo. No entanto, existem obstáculos, como paredes, que o Pac-Man não pode atravessar.

Além disso, fantasmas percorrem o labirinto com o objetivo de capturar o Pac-Man. A movimentação dos fantasmas também é modelada através do grafo, e estratégias podem ser implementadas para otimizar o desafio que eles representam.

Solução proposta

A solução proposta foi a criação do jogo Pacman a partir do algoritmo de Floyd-Marshall, um algoritmo utilizado para encontrar os menores caminhos em um grafo ponderado com arestas direcionadas ou não direcionadas, ou seja, que podem ou não possuir orientação associada a elas.

A escolha deste algoritmo deu-se por diversas razões, sendo algumas delas:

- a) Poder calcular os menores caminhos entre todos os pares de células no labirinto, permitindo um cálculo mais eficiente das rotas dos fantasmas para alcançar o pac-man;
- b) Poder calcular o caminho mais curto para que o Pac-Man alcance itens, e
- c) Poder analisar as movimentações em tempo real e poder ajudar na detecção de áreas onde os fantasmas podem se tornar presos ou onde o Pac-Man pode estar em perigo.

Desta forma, o algoritmo escolhido apresenta muitas vantagens em relação ao cálculo de rotas quando posto lado a lado com outros algoritmos de mesmo fim.

Diagrama

O diagrama de fluxo logo abaixo representa a lógica de funcionamento do jogo implementado em C++ envolvendo o Pacman, o Fantasma (Ghost), e o labirinto. A seguir vamos detalhar cada parte do diagrama:

1. Inicialização:

Nesta fase, o gerador de números aleatórios é iniciado para ser usado posteriormente no movimento automático do Fantasma.

2. Criação do Labirinto:

Um labirinto é criado, representado como uma matriz. Paredes são adicionadas ao labirinto para criar obstáculos.

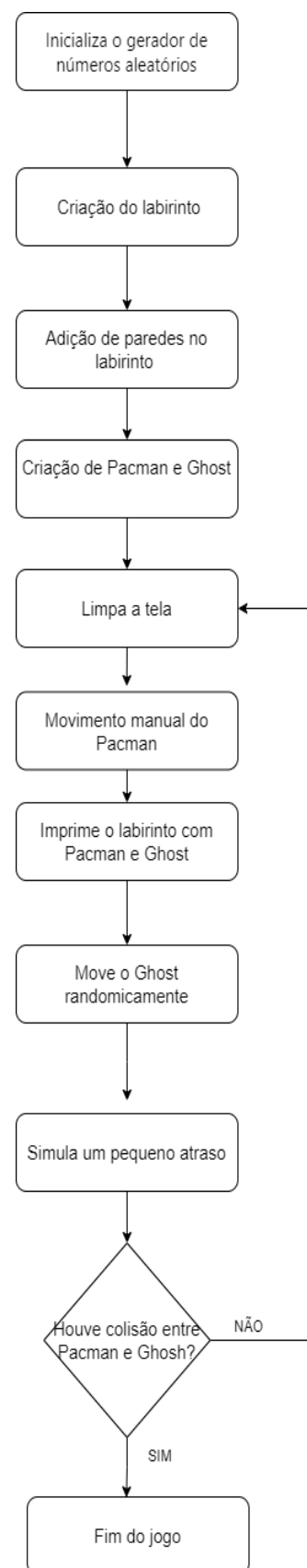
3. Criação de Pacman e Ghost:

Instâncias dos personagens Pacman e Fantasma são criadas e posicionadas inicialmente no labirinto.

4. Loop Principal:

O jogo entra em um loop principal que executa continuamente até que ocorra uma condição de término (colisão entre Pacman e Ghost).

- A cada iteração do loop, a tela é limpa (para sistemas Unix-like);
- O movimento do Pacman é controlado manualmente pelo usuário. Isso implica que o Pacman se move para uma nova posição conforme decidido pelo usuário;
- A posição atual de Pacman e Ghost é impressa no labirinto. Paredes são representadas por '#' e espaços livres por '.'.
- O Fantasma se move randomicamente em cada iteração do loop, criando um elemento de imprevisibilidade no jogo.
- Uma pausa curta é simulada para representar o tempo de processamento.



5. Condição de colisão:

A cada iteração do loop, verifica-se se o Pacman e o Ghost estão na mesma posição. Se isso ocorrer, o jogo imprime "Game Over" e termina.

6. Fim do Jogo:

Se uma colisão entre Pacman e Ghost for detectada, o jogo exibe "Game Over" e encerra.

Se o Pacman chegar no T(fim do jogo), o jogo exibe "Vitória".

Este diagrama de fluxo ilustra a sequência de eventos no código, destacando a interação entre os personagens e o ambiente do jogo. Ele oferece uma visão clara do fluxo de controle e das decisões tomadas durante a execução do programa.

