

1 Descrição da Aplicação

As Visual Novels são jogos interativos, na qual o usuário pode fazer diversas decisões, podendo elas levarem a finais diferentes por caminhos parecidos, ou finais iguais por diferentes caminhos. Atualmente, alguns dos exemplos mais famosos são: *Doki Doki Literature Club*, para PC, *The Dark Pictures: Man of Medan*, para PS4 e *Underworld Office* para mobile.

Esta categoria de jogos entretém o usuário por ser uma simulação da vida em um universo fictício. Assim como na vida real, toda decisão leva a uma consequência, e tudo está sob o controle de quem joga. Eles podem ser de diferentes gêneros: suspense, terror ou também romance e comédia.

O objetivo deste projeto foi desenvolver um jogo do tipo Light Novel, no qual o usuário assume o papel do protagonista e desenvolve a história por meio de suas escolhas. A versão feita para esse trabalho conta apenas com escrita em console, diferentemente das *Visual Novels* que contam com efeitos gráficos.

O título da Light Novel é **Everwinter Park**. A história gira em torno de um grupo de amigos que decide visitar um parque abandonado com o objetivo de encontrar algo sobrenatural e para comemorar o aniversário de uma das integrantes. O usuário assumirá o papel de Noah, a protagonista e aniversariante. Dependendo das escolhas feitas, a aventura pode ser tanto curta quanto longa, e, toda escolha leva a um resultado, seja ele bom ou ruim. O gênero da Novel é focado principalmente no terror, e, dependendo do caminho que seguir, drama e romance.

Para iniciar o jogo, só é necessário clicar *Enter* e ele direcionará o jogador ao arco inicial da história. A partir desse ponto, começarão as escolhas: pode haver de uma a quatro opções de resposta ou ação a se fazer. Para selecionar, é requerido digitar o número decisão. O jogo não tem opção de volta e nem salvar, pois assim como na vida real, não há opção nem de voltar no tempo e muito menos de parar ele. O único benefício dado ao jogador é um tempo infinito para poder tomar suas decisões com bastante cautela.

2 Abordagem do ponto de vista da disciplina

Dentre todas as estruturas de dados aprendidas durante o decorrer da disciplina, a que mais se adequou ao desenvolvimento da Light Novel foi a do Grafo, mais especificamente, a do **Grafo Direcionado** e sem pesos. A razão para essa escolha pode ser justificada, uma vez que para o funcionamento correto do jogo, assim como explicado na descrição, não deve ser possível retornar a um mesmo ponto da história utilizando o caminho de ida.

Os vértices do grafo são usados para acessar os blocos de história. Cada vértice recebe um identificador, que além de ser sua posição na lista de adjacências, também é o identificador dele em `vetor_nomes`, um vetor que armazena o nome dos arquivos de texto (*.txt*) correspondente a cada vértice do grafo, o que auxilia a organização das escolhas e a impressão no console.

Tratando-se da implementação, como já mencionado, foi usada a Lista de Adjacências em vez da matriz, pois a partir de um vértice, a lista permite o acesso à um vértice associado sem precisar usar algum mecanismo de busca mais complexo, e as únicas informações de real interesse são a quais vértices ele direciona.

As função que ganha mais atenção em todo o jogo é a função `void escolher(char **v, Grafo* g, int id)`, o mecanismo de escolha do jogo. Ela é uma função recursiva, encarregada de processar o *input* do usuário e levá-lo ao vértice escolhido. Para isso, é percorrida a lista de adjacências do vértice escolhido - o que não é tão custoso, já que como mencionado na descrição, cada vértice direcionará a no máximo 4 outros vértices. Ao encontrar o vértice escolhido, ele envia esse à função. Caso o vértice dado à função não direcione a nenhum outro, a história é considerada finalizada e o programa é encerrado.

Algumas funções auxiliares usadas foram:

- `void build(Grafo* g, char **v)`: cria o vetor que contém os nomes dos arquivos e monta o grafo via inserção de arestas.
- `void play(Grafo* g, char **v)`: inicializa a criação do grafo, imprime a tela inicial e roda a função `escolher`.
- `void imprime_texto(char *nome_arq)`: utilizando o nome do arquivo de texto encontrado no `vetor_nomes`, a função lê e imprime no console o conteúdo do arquivo.

