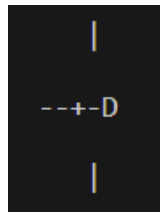


TRABALHO 02

Objetivo do Trabalho: Reimplementar o algoritmo do jogo escolhido no trabalho 01 de forma orientada a objetos e fazer um vídeo explicando o que mudou. Além disso, foi elaborado o diagrama referente ao jogo e também foi feita a descrição do mesmo.

Jogo Escolhido: Jogo de Avião - FlySim

Representação do Avião:



Representação dos Obstáculos: # - Linux

■ - Windows - Código Ascii 219;

Representação da Borda do Jogo: = Código Ascii 205

Descrição do Algoritmo: O algoritmo implementa um jogo de avião onde o usuário controla um avião para evitar colisões com blocos em branco e assim alcançar o maior número de pontos possíveis. O jogo em si, acontece através de um loop principal, que utiliza uma thread para controlar os obstáculos que surgem em meio ao terminal e outra thread para controlar a movimentação do avião por meio dos obstáculos. A posição dos obstáculos e do avião é atualizada continuamente na tela do terminal.

Classes, Funções Utilizadas e o que Fazem

Base.hpp: Esta classe contém funções utilizadas para verificar quais teclas o usuário clicou e posteriormente a partir destas teclas o avião é movimentado na tela do terminal. Além disso, nesta classe também estão sendo definidos os caminhos que o avião poderá percorrer, está definido o tempo que irá ser atualizado à tela, à função de limpar a tela e também possui a definição dos caracteres que

representam a borda do jogo e o bloco que representa o obstáculo que deverá ser desviado.

Block.hpp: Nesta classe estão sendo definidos o bloco que representa o obstáculo, seu tamanho em x e y, à função que irá implementar este bloco posteriormente e também uma função para o imprimi-lo na tela do terminal.

Block.cpp: Já nesta é incluído à classe Block.hpp e posteriormente é implementado as funções que foram previamente definidas na classe que foi incluída.

Level.hpp: Nesta classe são incluídas as classes Block.hpp e Plane.hpp, são definidos blocos vazios e com obstáculos, é definido o avião, o nível do jogo, sua dificuldade e também é definido se o jogo está sendo executado e qual à tecla que o usuário pressionou. Já as funções presentes nesta classe definem as impressões de obstáculos, à atualização da localização do avião na tela, o nível do jogo, à impressão da borda, à execução do jogo e também serve para definir o controle do usuário.

Level.cpp: Já esta classe implementa todas as funções que foram previamente definidas na classe Level.hpp.

Plane.hpp: Nesta classe estão sendo definidos à localização em x e y do avião, um block que representa o avião enquanto está ocorrendo o jogo e quando o usuário bate em um obstáculo, já as funções definem as posições atuais, o avião no jogo, o avião quando bate em um obstáculo e também se houver mudança da posição do avião por parte do usuário.

Plane.cpp: Já nesta classe estão sendo implementadas todas as funções que foram previamente definidas na classe Plane.hpp.

Main.cpp: Esta é a classe principal do jogo, nela estão sendo incluídas todas as classes citadas anteriormente. Estão sendo utilizadas duas threads, a primeira serve para controlar os obstáculos, seu nível de dificuldade e a pontuação do jogador, já a segunda serve para controlar as teclas que foram pressionadas pelo usuário e posteriormente movimentar o avião para a direção escolhida. Ademais, o jogo é atualizado continuamente enquanto o usuário do jogo não colide o avião com um obstáculo.

Como funciona o Jogo: O jogo funciona de maneira interativa, onde o objetivo do usuário é desviar dos obstáculos do jogo, utilizando-se das teclas W, A, S e D e por consequência tentar alcançar o maior número de pontos possíveis (score).

Explicação do Jogo: O jogo é bem simples, após executá-lo é impressa uma borda para delimitar o jogo. Posteriormente, os obstáculos e o avião são impressos de forma aleatória no terminal e o score do jogador vai aumentando conforme o nível de dificuldade do jogo. A partir daí, o usuário terá que desviar o avião dos obstáculos através das teclas: W, A, S e D, que servem respectivamente para movimentar o avião para cima, para à esquerda, para baixo e para à direita. Se o usuário deixar o avião bater em algum obstáculo o jogo é encerrado e é impresso a sua pontuação final.

Como faz para executá-lo: Para executar o jogo basta executar seu arquivo (game.exe em Windows ou Game em Linux) em um editor de código fonte, por exemplo Visual Studio Code.