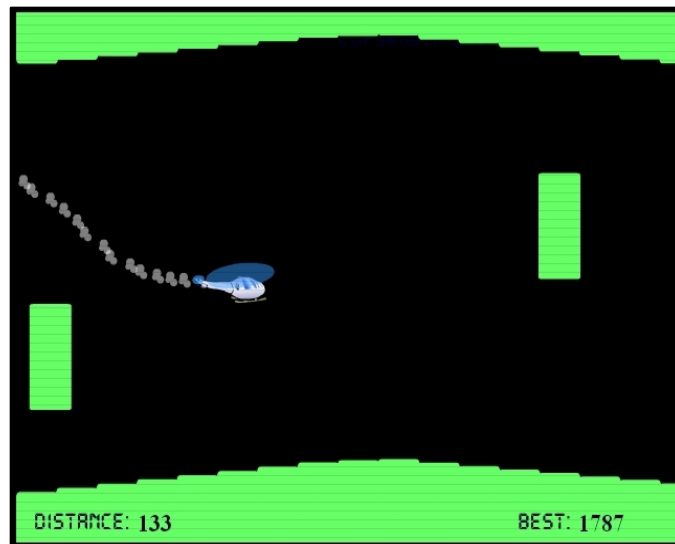


Especificação do Projeto Copter



Turma 6-Grupo 11

Maria João Mira Paulo – up201403820
Nuno Ramos- up201405498

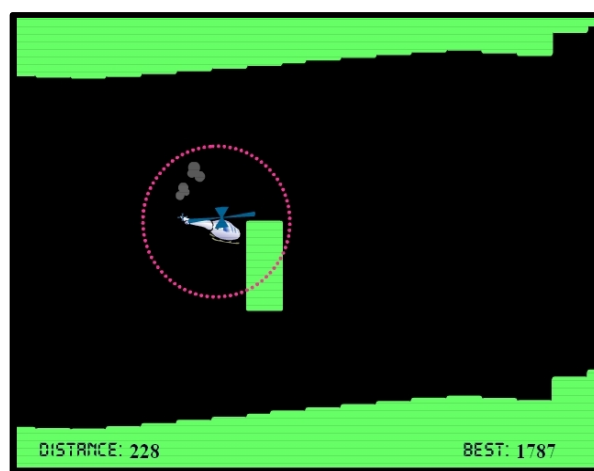
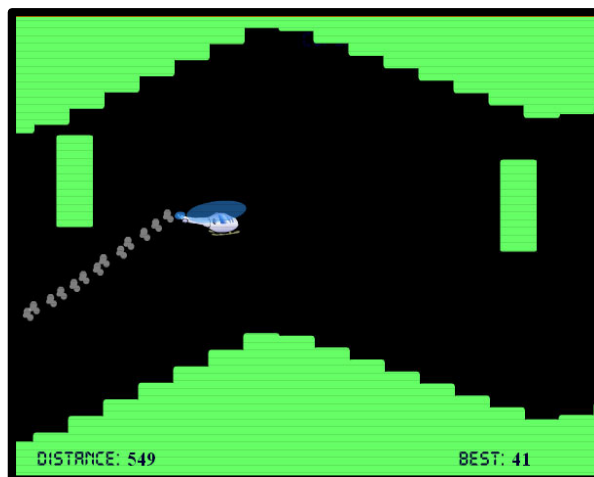
Descrição

A ideia baseia-se num jogo de computador cujo objetivo é desviar um helicóptero de eventuais obstáculos que surgem no ecrã, durante a maior distancia possível (a considerar o tempo total da tentativa).

O helicóptero só pode voar na parte preta do ecrã e quando toca num obstáculo ou numa parede verde, cai, perdendo o jogo.

O jogador só pode pilotar o seu helicóptero para cima ou para baixo. Tem duas opções, ou usar o rato ou usar o teclado para manter o helicóptero em voo.

Quando pressiona continuamente o rato/tecla (provavelmente a tecla espaço) o helicóptero sobe. Quando a solta o helicóptero desce. Para o manter no mesmo sitio o jogador tem de fazer cliques consecutivos no rato ou na tecla.



Periféricos a utilizar

Os dispositivos de E/S que pretendemos usar são:

- **Placa Gráfica**, utilizada para criar a interface com o utilizador;
- **Timer**, utilizado nas animações, interrupções e lógica do jogo;
- **Rato**, utilizado na lógica do jogo, por exemplo o Menu;
- **Teclado**, utilizado na lógica do jogo;
- **RTC**, para retornar o tempo real;

Para lidar com os periféricos vamos usar interrupções.

Módulos a desenvolver

- **Main.c**: para entrar e sair do modo gráfico e chamar as principais funções do programa;
- **Copter.c e Copter.h**: implementação de uma estrutura relativa ao helicóptero (coordenadas x e y onde se encontra, altura, largura, imagem), e principais funções como criar, eliminar, desenhar e atualizar helicóptero.
- **Obstacles.c e Obstacles.h**: implementação de uma estrutura relativa aos obstáculos (coordenadas x inicial, x final, y inicial, y final) e principais funções como criar e eliminar obstáculo.
- **Margins.c e Margins.h**: implementação de uma estrutura relativa as margens verdes do ecrã, com função que permita atualizar margens.
- **GameState.c e GameState.h**: implementação da classe principal do jogo, com funções relativas à logica de jogo como Game Over, Update Game, Copter Hit Margine, Copter Hit Obstacle.
- **VBE.c e VBE.h**: implementação das VESA BIOS Extensions.
- **KBC.c e KBC.h**: implementação de funções que leem e escrevem no KBC.
- **Keyboard.c e Keyboard.h**: implementação de funções para manipular o teclado.

- **Mouse.c e Mouse.h:** implementação de funções para manipular o rato.
- **Timer.c e Timer.h:** implementação de funções para manipular o contador.
- **RTC.c e RTC.h:** implementação de funções para manipular o RTC.
- **Menu.c e Menu.h:** implementação de uma estrutura relativas ao Main Menu com um apontador para uma imagem, FALTA e funções relativas a iniciar jogo, sair, escolher helicóptero.

Plano de Desenvolvimento

Na **primeira semana** pretendemos desenvolver a parte gráfica base do jogo: construção do helicóptero, fazendo o mover-se para cima sempre que é clicado a tecla do espaço e para baixo sempre que é largada a tecla do espaço e iniciar a implementação de um algoritmo que desenhe as margens aleatoriamente, ao longo do jogo.

Na **segunda semana** pretendemos terminar a implementação de um algoritmo que desenhe as margens aleatoriamente, ao longo do jogo e implementar funções que desenhem os obstáculos no ecrã ao longo do jogo.

Na **terceira semana** pretendemos desenvolver a lógica de jogo, sempre que o helicóptero vai contra um obstáculo ou contra as margens, cai, perdendo o jogo.

Na **quarta semana** pretendemos desenvolver o menu do jogo e melhorar a interface gráfica. Pretendemos ainda implementar o periférico RTC, mais especificamente o aparecimento da data atual no ecrã de jogo.

Na **quinta semana** pretendemos melhorar a interface gráfica, testar e corrigir possíveis erros do programa.

Possíveis Melhorias:

Caso exista oportunidade pretendemos ainda implementar um modo de jogo com multi-jogadores, mais concretamente a possibilidade de aparecimento de vários helicópteros na mesma tela de jogo, controlados por teclas diferentes.

Além disso, pensamos em implementar a possibilidade de o jogador escolher o próprio ícone do “copter”, como um avião de papel, um pássaro, nave espacial, etc.