

# Copter

Laboratório de Computadores

Turma 6 - Grupo 11



Maria João dos Santos Aguiar e Mira Paulo

Nuno Miguel Mendes Ramos

2ºano MIEIC



## Índice

1. RESUMO DO JOGO.....	3
2. DESCRIÇÃO DO PROGRAMA .....	4
MENU PRINCIPAL.....	4
MENU DE OPÇÕES .....	5
GRÁFICOS POSSÍVEIS DO JOGO .....	6
MENU DE “GAME OVER” .....	8
CRÉDITOS.....	9
3. PERIFÉRICOS USADOS .....	10
TIMER.....	10
TECLADO.....	10
RATO .....	11
PLACA GRÁFICA .....	11
RTC.....	11
4. DESCRIÇÃO DA ARQUITETURA DO PROGRAMA.....	13
MÓDULOS E DESCRIÇÃO.....	13
5. ESTADO DO PROJETO.....	16
INSTALAÇÃO .....	16
POSSÍVEIS MELHORIAS .....	16
EMPENHO DOS ELEMENTOS DO GRUPO .....	16
6. GRÁFICO DE INVOCAÇÃO DE FUNÇÕES .....	18
7. BIBLIOGRAFIA.....	20



## 1. Resumo do Jogo

O copter é um jogo que tem um principal objetivo a pilotagem de um helicóptero através de um labirinto de obstáculos. É um jogo bastante simples graficamente, que permite ao utilizador jogar com o teclado ou com o rato. Então, para fazer subir o helicóptero o jogador pode, ou pressionar a tecla do espaço, ou pressionar o botão esquerdo do rato.

O objetivo do jogo é atingir a maior distancia possível sem bater em numa das margens ou em nenhum dos obstáculos que aparecem com o passar do tempo.



## 2. Descrição do Programa

### Menu Principal



O menu principal do jogo é constituído por 4 botões. O botão "Play" que inicia um novo Jogo, um botão "Exit" que sai do Jogo, um botão "Options" que possibilita ao jogador a escolha do seu "copter" e um botão de "Credits".



## Menu de Opções



O jogador pode assim alterar o ícone do jogo. Ao clicar em “Home” terá acesso de novo ao menu principal onde pode começar a Jogar.

### Estratégia Utilizada

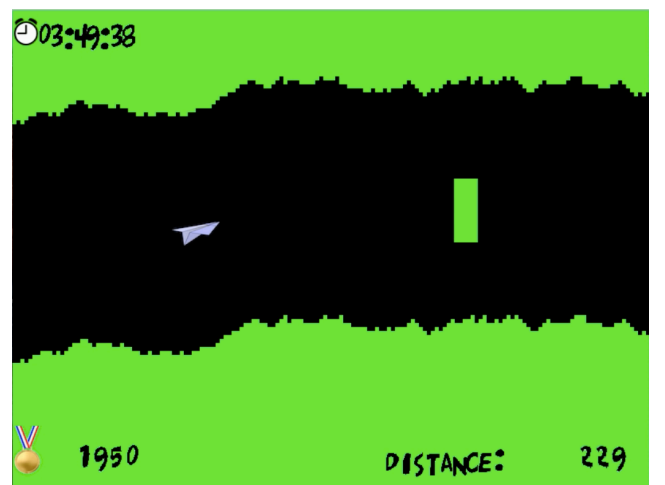
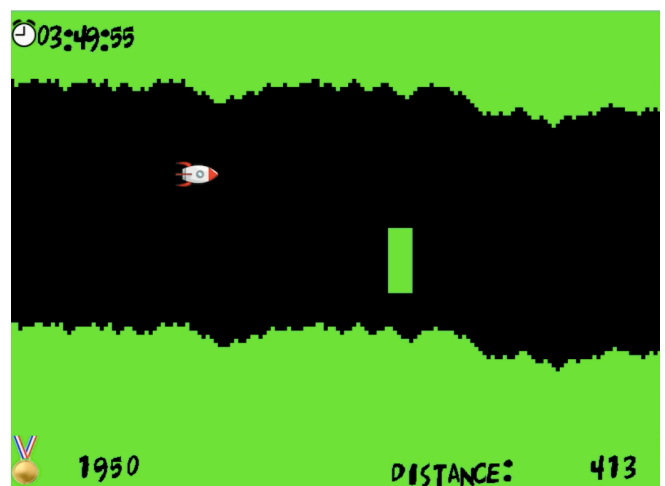
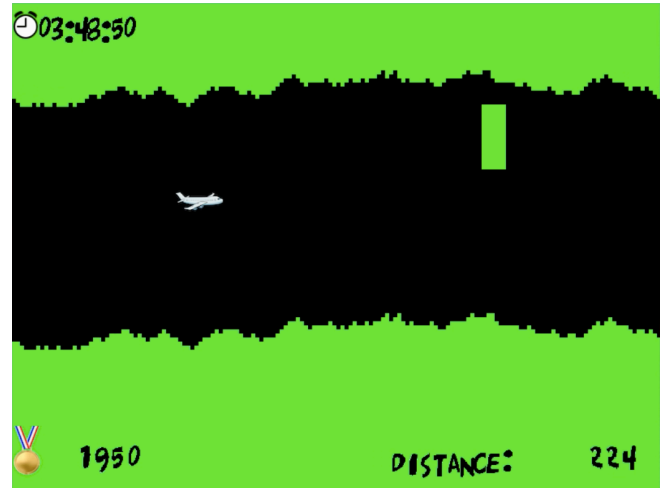
A estratégia utilizada na criação do Menu de Opções foi a criação de uma imagem constituída por 6 pequenas imagens de possíveis helicópteros que distanciam 60 pixéis umas das outras.

Assim, foi criada uma função que recebe um apontador para Bitmap, uma coordenada x e uma coordenada y onde vai ser desenhada a imagem e um char, que significa a imagem respetiva, '0' caso seja a imagem zero, '1' caso seja a imagem um e assim sucessivamente. Seguidamente, o que essa função faz é subtrair o código aschii do char que recebe como argumento com 48 (código aschii do 0), obtendo se assim o número de imagens que se salta. Esse número multiplicado por 60 origina o número de pixéis que o apontador para o início da imagem se terá de descolar, ficando assim a apontar para a imagem pedida.



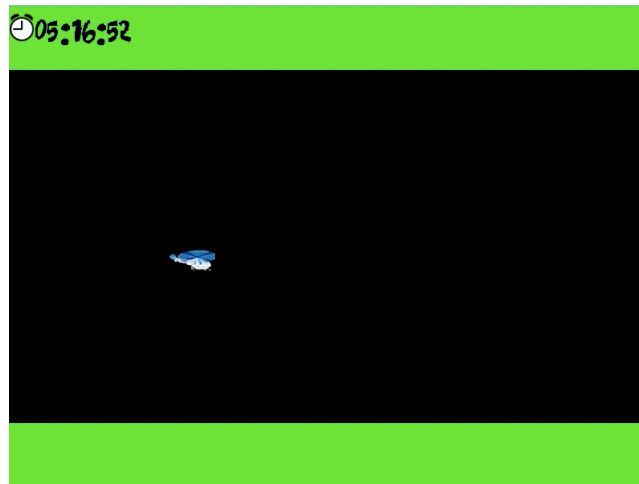


## Gráficos possíveis do Jogo

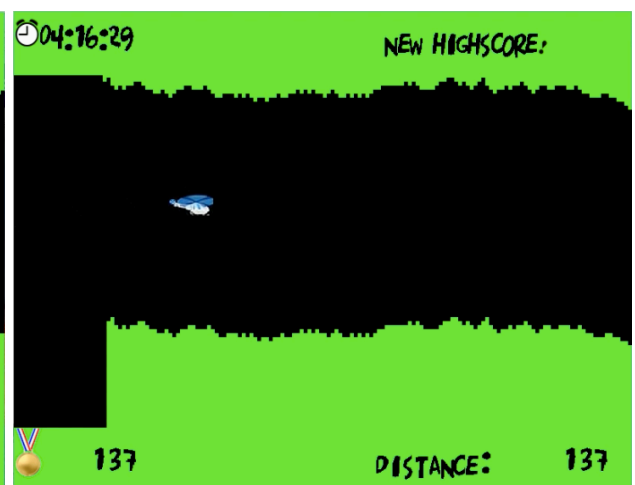
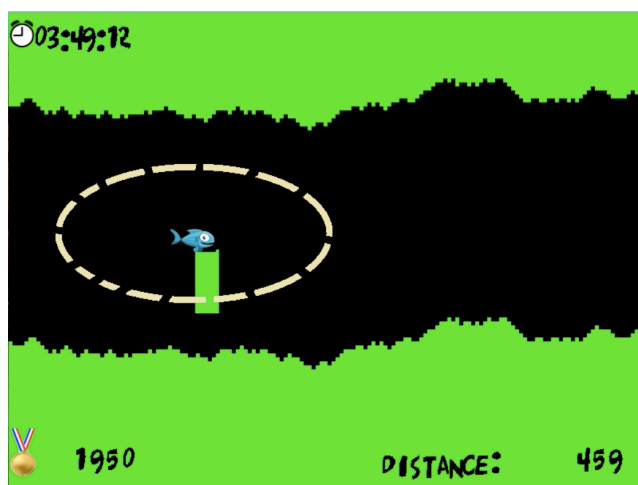




Quando se inicia o jogo, o helicóptero entra num modo de pausa até ser pressionada a minha vez a tecla do espaço.



Além de ser possível trocar a imagem relativa ao helicóptero, o jogador pode ainda ter acesso ao seu "HighScore", à data atual e à distância percorrida durante o jogo. Sempre que um jogador atinge uma nova "HighScore" uma mensagem aparecerá no lado superior direito do ecrã a avisar do seu novo recorde. Sempre que o jogador perde, ao conduzir o helicóptero para uma margem ou um obstáculo, um círculo aparecerá no local onde o helicóptero embateu.





## Menu de “Game Over”



Assim que o jogador perde, é direcionado para um Menu de “Game Over”, onde pode, ou jogar novamente, ou voltar para o menu principal ou sair completamente do jogo.





## Créditos



O jogador pode voltar ao menu principal ao pressionar o botão “Home”.



### 3. Periféricos Usados

Periférico	Interrupção
Timer	Sim
Teclado	Sim
Rato	Sim
Placa Gráfica	Não
RTC	Não

#### Timer

Principal função do timer é a construção dos "frames", é também usado para atualizar o helicóptero: as suas coordenadas e a distância alcançada, e para regular o tempo das imagens, por exemplo na "highScore", sempre que avisa o jogador de um Novo "HighScore".

#### Teclado

É usado na lógica principal do jogo. Para fazer o helicóptero subir o jogador deve pressionar a tecla do espaço, e para fazer descer, deve largá-la.



## Rato

É usado na lógica principal do jogo. Para fazer o helicóptero subir o jogador pode, além de pressionar a tecla do espaço, pressionar o botão do lado esquerdo do rato e larga-lo para o fazer descer.

Além disso é usado em toda a parte relativa aos menus do Jogo: menu principal, menu de opções e menu de Game Over.

## Placa Gráfica

Utilizada para desenhar todos os bitmaps utilizado no programa, tanto no jogo, como nos menus. É utilizado o modo vídeo 0x114, de resolução 800x600. É utilizado o double buffer. As colisões são detetadas por cores. As Fontes, são usadas através dos bitmaps.



Foi criada uma imagem com o alfabeto ordenado segundo os seus códigos aschii, sendo que a distancia entre dois char foi declarada como 20 pixéis. Assim, foi criada uma função que recebe um apontador para Bitmap, uma coordenada x e uma coordenada y onde a partir de onde é desenhada a imagem e um char, neste caso o char correspondente à letra, número ou símbolo que queremos imprimir. Como sabemos que o código aschii do primeiro elemento é 48, ó código apenas trata de subtrai o código aschii do char que queremos imprimir a 48. O resultado será o número de letras a frente do zero, primeiro elemento. Se multiplicarmos esse resultado por 20 e somarmos ao apontador para o inicio da imagem obtemos um apontador para o inicio letra que queríamos imprimir.



## **RTC**

Utilizado sempre que o jogador inicia uma tentativa de jogo, ao mostrar no canto superior esquerdo as horas atuais.



## 4. Descrição da Arquitetura do Programa

### Módulos e Descrição

- **Bitmap**

Responsável por desenhar todas as imagens do programa, por exemplo desenhar imagens com e sem fundo.

- **Main**

Responsável por chamar as funções de subscribe e unsubscribe, vg\_init e vg\_exit e a chamar a função principal do programa.

- **Button**

Criação de um botão e de uma função que testa se este foi ou não pressionado)

- **Copter**

Criação de um helicóptero e funções de atualização do mesmo, por exemplo a posição.

- **GameState**

Contém funções que atualizam os dados do helicóptero, e permitem escrever a cada novo "frame", os dados já atualizados.

- **HighScore.txt**

Ficheiro que contém a melhor "highScore".

- **KBD**



Contém grande parte das constantes presentes no programa.

- **Keyboard**

Responsável pelas funções que se relacionam com o teclado.

- **MainMenu**

Contém todos os Menus existentes no programa, assim como as funções que se relacionam com cada menu (construtor, destrutor, carregamento de imagens, etc).

- **Margins**

Responsável por desenhar as margens do programa.

- **Menu**

Funções de subscribe, unsubscribe, carregamento e remoção de imagens.

- **Mouse**

Responsável pelas funções que se relacionam com o rato.

- **Obstacles**

Responsável por desenhar os obstáculos do programa.

- **Rtc**

Responsável pelas funções que se relacionam com o rtc.

- **RTCC**

Contém constantes relativas ao rtc.

- **Singleplayer**

Contém o ciclo principal que corre o jogo.



- **Timer**

Responsável pelas funções que se relacionam com o timer.

- **vbe**

Responsável pelas funções que se relacionam com a placa gráfica.

- **vídeo\_gr**

Responsável pelas funções que se relacionam com o carregamento e tratamento de imagens e frames.

## Estruturas Criadas

- Singleplayer
- Copter
- MouseInfo
- Obstacles
- Button
- MainMenu
- MenuGameOver
- OptionsMenu



## 5. Estado do Projeto

### Instalação

Para iniciar o programa basta entrar na pasta `proj` e, como super user (su) escrever os seguintes comandos: **sh install.sh** (copia os ficheiros de imagem para um ficheiro específico e compila o jogo), e depois **sh run.sh** (correr o programa).

### Possíveis Melhorias

Como previsto, não foi implementado o periférico da porta de série.

Embora tenha sido referido na especificação do projeto a opção de jogo de “Multiplayer”, não foi possível a sua implementação. Apercebemo-nos que seria impossível fazer subir os dois helicópteros ao mesmo tempo, pois como só é possível ler um “scancode” de cada vez, este modo de jogo só funcionaria se os dois jogadores pressionassem nas teclas desfasadamente, o que deixaria de fazer sentido.

### Empenho dos Elementos do Grupo

O grupo envolveu se igualmente no trabalho.





## **Detalhes de Implementação**

As aulas de Lcom foram bastante benéficas para a aprendizagem dos periféricos e, por isso bastante úteis para o projeto. Pensamos, no entanto, que tanto o RTC como a porta de série deveriam ter sido mais explorados durante as aulas.

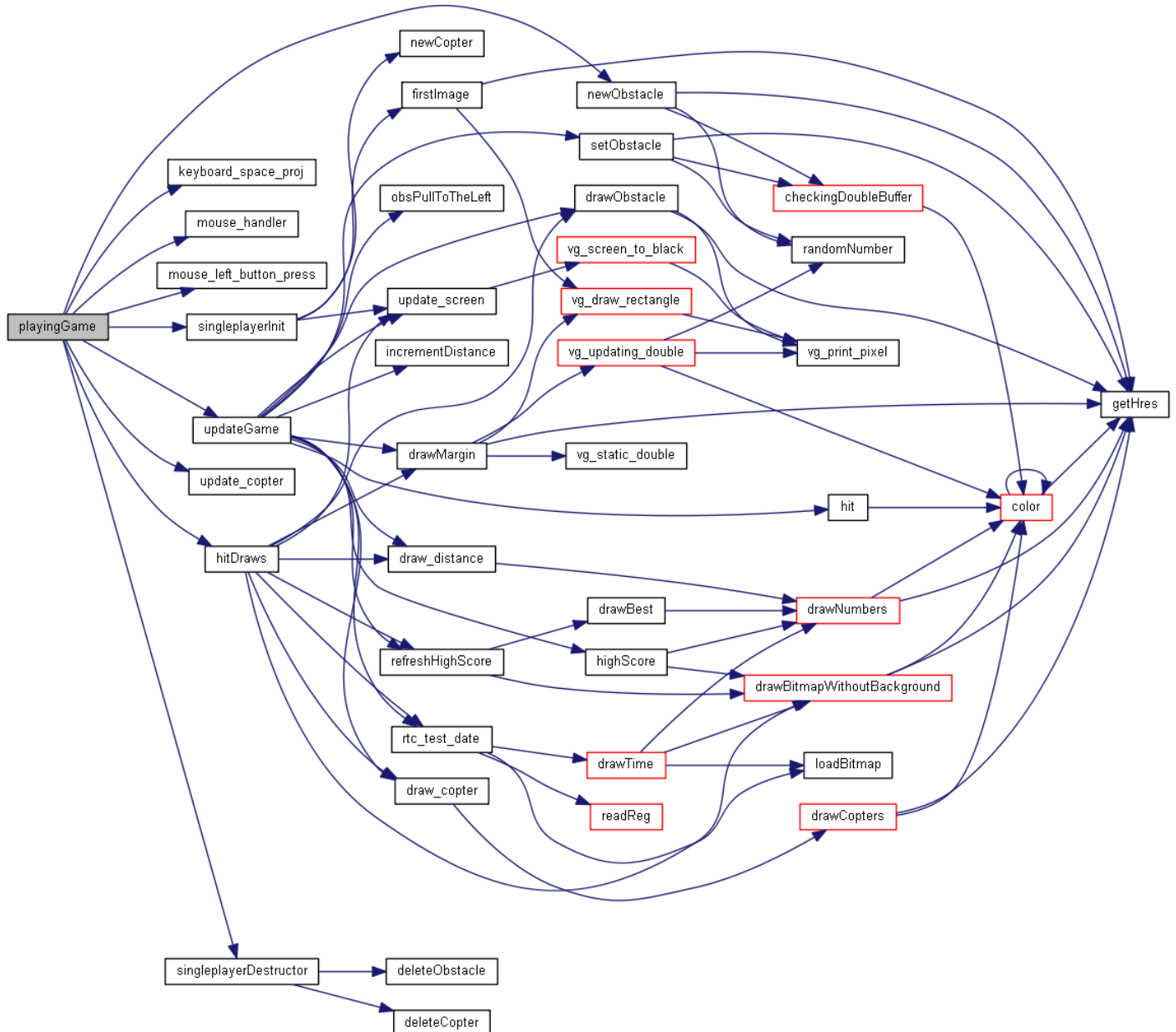
## **Dificuldades Sentidas**

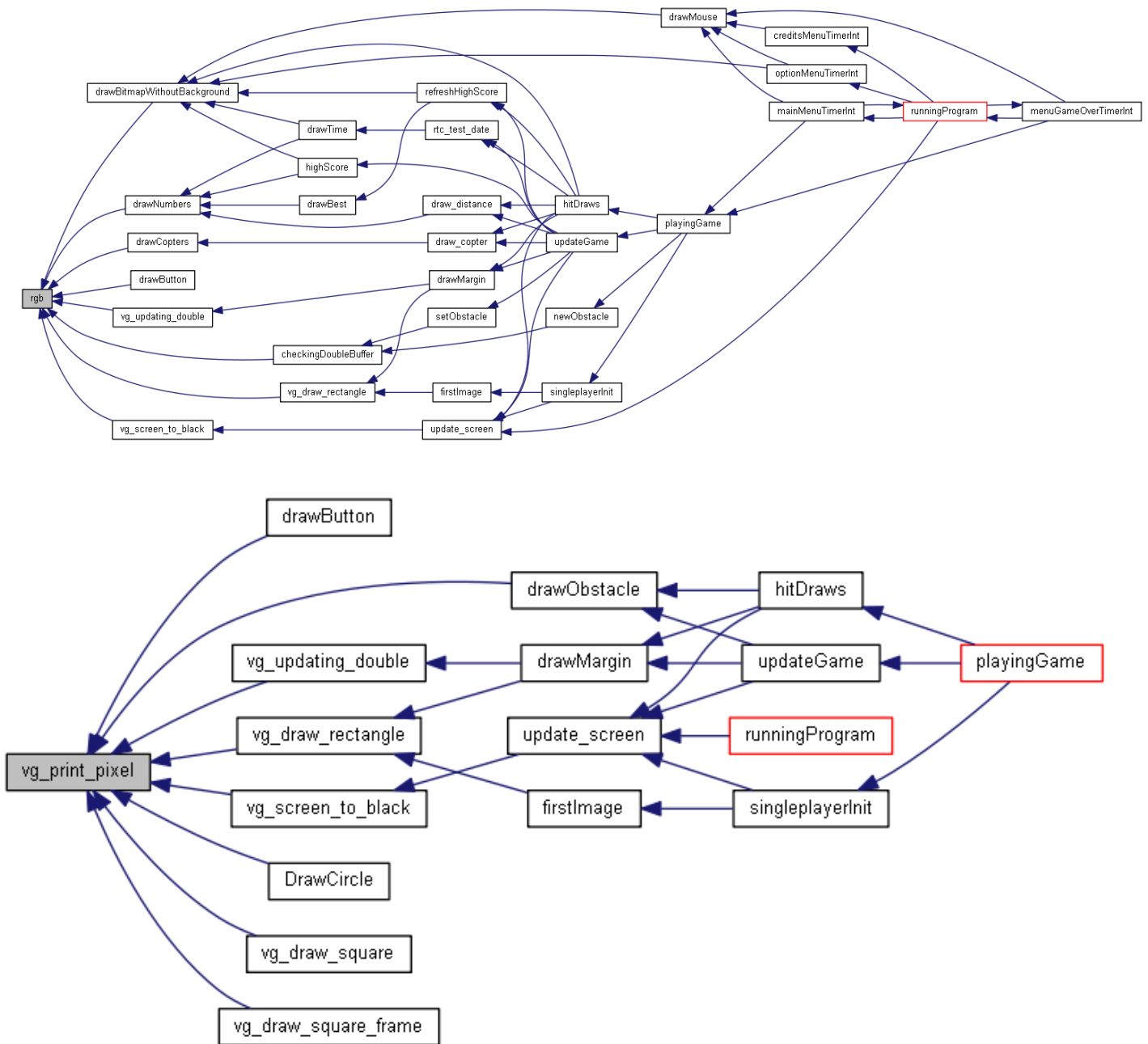
Deparámo-nos com bastantes dificuldades no que diz respeito à gestão de memória e também na implementação dos bitmaps.

A criação das margens foi também uma tarefa complicada. Inicialmente tínhamos uma estrutura margens mas, por estar trazer bastantes problemas no jogo, decidimos fazer uma implementação sem objetos margens, escrevendo diretamente no vídeomem tentando manter a lógica do jogo, implementação essa que resultou.



## 6. Gráfico de Invocação de Funções







## 7. Bibliografia

O código relativo aos bitmaps (vídeo\_gr), mais especificamente a função de carregamento de imagens foi inspirada do blog do Henrique Ferrolho: <http://difusal.blogspot.pt/2014/09/minixtutorial-8-loading-bmp-images.html>, baseado na página <http://forums.fedoraforum.org/archive/index.php/t-171389.html>.