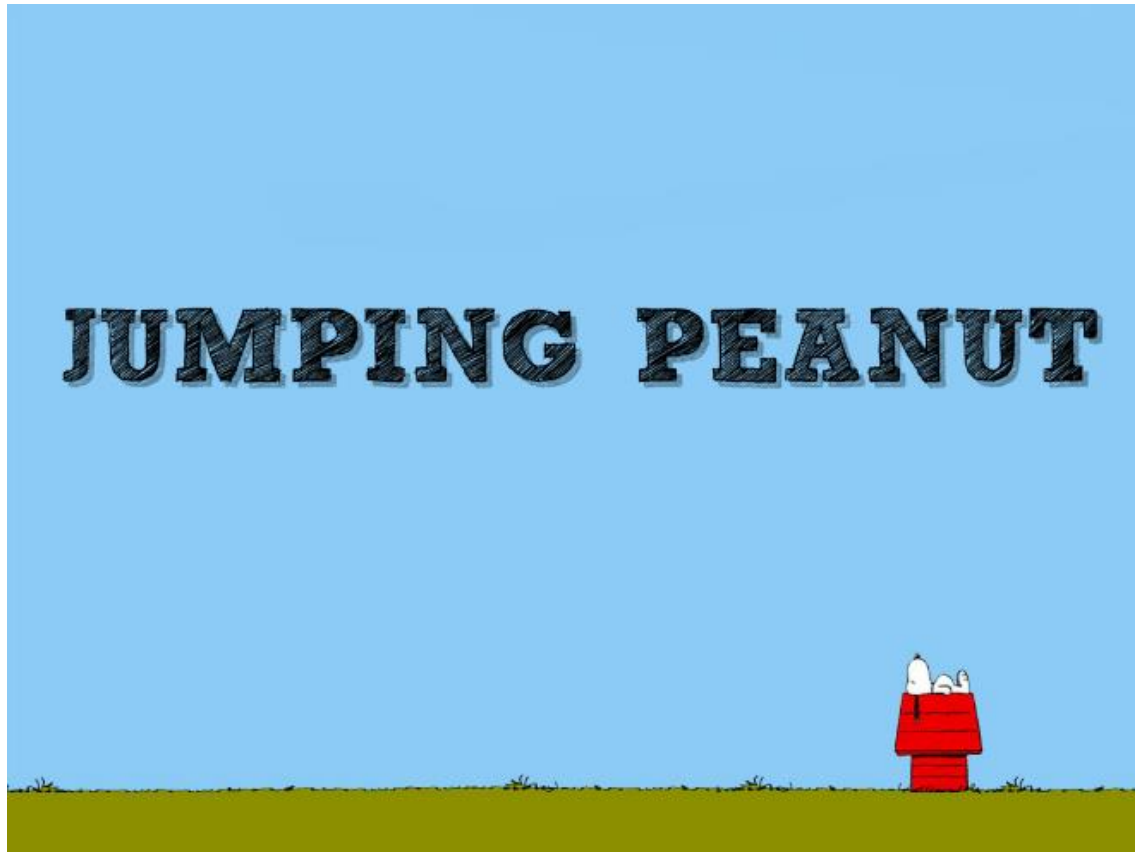




**FEUP** FACULDADE DE ENGENHARIA  
UNIVERSIDADE DO PORTO



*Laboratório de Computadores 2017/2018*

*Turma 7 Grupo 3*

Diogo Yaguas up201606165@fe.up.pt

Maria Viana up201604751@fe.up.pt

*2 de janeiro de 2018*

# Índice

|  |    |
|--|----|
| Instruções de utilização do programa <sup>[1]</sup> <sub>SEP</sub> ..... | 3  |
| Menu Principal .....   | 3  |
| Jogo.....  | 4  |
| Game Over Menu .....   | 6  |
| Periféricos.....   | 7  |
| Timer .....  | 7  |
| Teclado .....  | 7  |
| Rato.....  | 7  |
| Placa Gráfica.....   | 8  |
| Real Time Clock.....   | 8  |
| Organização de código e estrutura .....                                  | 9  |
| Módulos .....  | 9  |
| <i>bitmap.c</i> .....  | 9  |
| <i>cloud.c</i> .....   | 9  |
| <i>game.c</i> .....  | 9  |
| <i>i8042.h/ i8254.h</i> .....  | 9  |
| <i>jumpingPeanut.c</i> .....   | 9  |
| <i>keyboard.c</i> .....  | 9  |
| <i>main.c</i> .....  | 10 |
| <i>mouse.c</i> .....   | 10 |
| <i>rtc.c</i> .....   | 10 |
| <i>stateMachine.c</i> .....  | 10 |
| <i>timer.c</i> .....   | 10 |
| <i>vbe.c</i> .....   | 10 |
| <i>video_gr.c</i> .....  | 10 |
| Diagrama de chamada de funções .....                                     | 11 |
| Detalhes da implementação.....   | 12 |
| Avaliação da unidade curricular.....                                     | 12 |
| Instruções de Instalação .....   | 12 |

## Instruções de utilização do programa

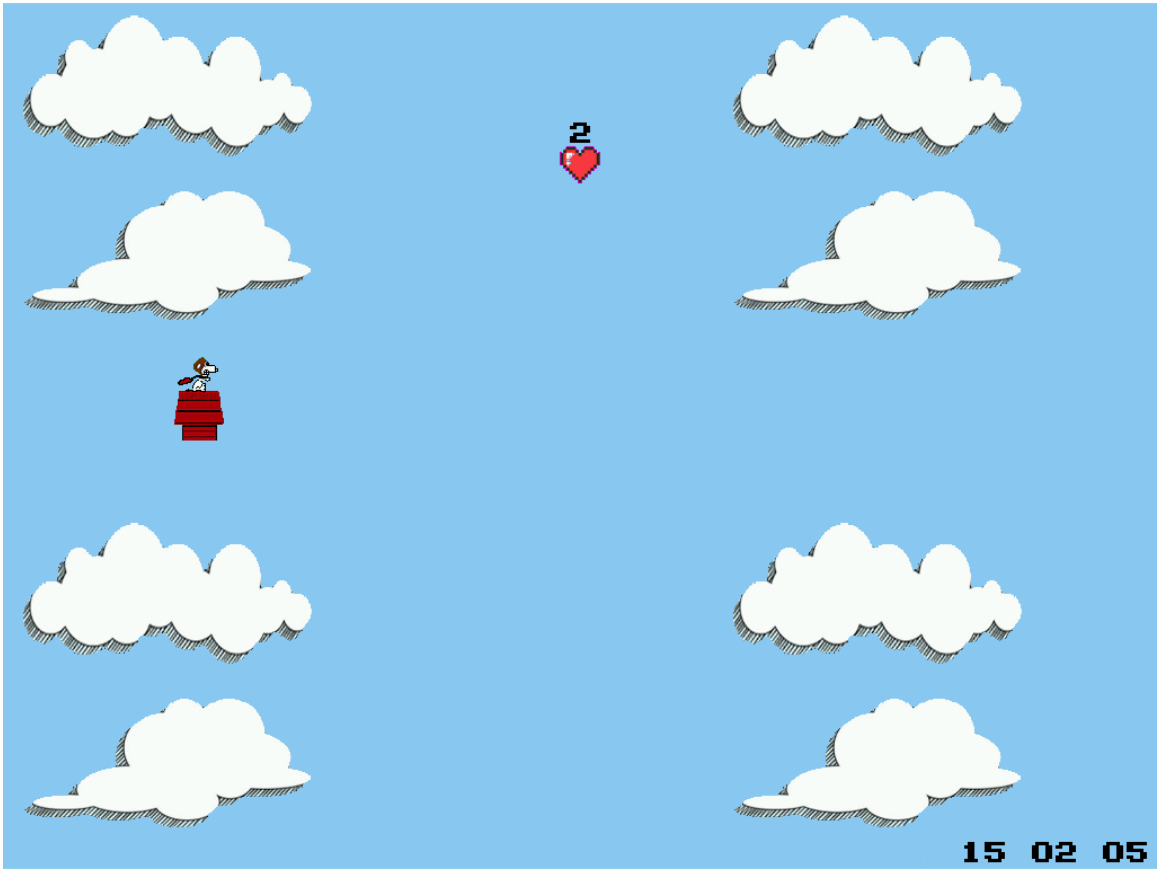
### Menu Principal

Ao correr o programa é apresentado inicialmente o menu principal do jogo em que o jogador tem duas alternativas que pode optar movendo o rato até cada uma delas e seleccionar a opção desejada com o botão esquerdo do rato: o Play, para iniciar um novo jogo e o Exit, para sair do programa. Pode também sair do menu pressionando o ESC e iniciar o jogo pressionado P.



## Jogo

O objetivo do jogo é evitar as nuvens, que são os obstáculos que vão aparecer, durante o máximo de tempo possível, usando o teclado e o rato para controlar a personagem. Para que este “salte” e suba, basta premir o espaço ou então usar o botão esquerdo do rato e arrastar para cima. O score vai sendo incrementado em uma unidade por cada lote de nuvens passadas, sendo que se bate em alguma ou sai fora do ecrã de jogo perde uma vida, que neste caso existem 3 e estas sempre que isto acontece são decrementadas em uma unidade. O score máximo que o jogador pode obter é de 999. A cada 25 pontos que o jogador alcança, se não tiver o número máximo de vidas, ganha uma vida.



## Game Over Menu

Este menu aparece quando o utilizador perde todas as vidas e, consequentemente, perde o jogo. Possui duas opções, que tal como no menu principal são escolhidas com o rato: o Try Again, que leva o jogador a iniciar um novo jogo; e o Exit, para sair do programa.



## Periféricos

| Dispositivo     | Utilização   | Interrupções |
|-----------------|--|--------------|
| Timer           | Atualização do estado de jogo                      | Sim          |
| Teclado         | Controlo da personagem e interação com os menus    | Sim          |
| Rato            | Navegação nos menus e controlo da personagem       | Sim          |
| Placa Gráfica   | Visualização do jogo, menus e imagens              | Não          |
| Real Time Clock | Alterar o fundo que é desenhado dependendo da hora | Não          |

### Timer

A principal função do timer é a atualização do estado de jogo, como atualizar as coordenadas dos obstáculos. Na função `updateJumpingPeanut()`, que está implementada no `jumpingPeanut.c` é onde se dá o processamento de todas as interrupções.

### Teclado

O teclado é usado para o controlo da personagem, fazendo com que esta salte quando o espaço é pressionado. Também pode ser usado como um atalho para os menus.

### Rato

Este é o dispositivo que é usado para navegar nos menus do jogo, em que é usado o botão esquerdo para selecionar a opção desejada. Estes menus estão todos implementados no módulo `game.c`. O rato pode também ser utilizado para a personagem saltar, pressionando o botão esquerdo e arrastando o rato para cima.

## Placa Gráfica

Este é o dispositivo é usado para apresentar a informação gráfica do jogo no modo gráfico 0x117. Todas as imagens são visualizadas com recurso À biblioteca bmp de Henrique Ferrolho, da qual fazem parte os ficheiros bitmap.c e bitmap.h; Já as imagens são da autoria dos autores deste projeto. Estas imagens são usadas em modo RGB (5:6:5).

Decidimos implementar um double buffer. Implementamos também duas imagens da personagem e alternámo-las para parecer que o cachecol que esta usa está a voar: isto foi implementado no drawJumpingPeanut() no módulo jumpingPeanut.c.

## Real Time Clock

O RTC é usado para ler a data e hora atual, a qual depois é desenhada no modo de jogo. Usamos também este dispositivo para alterar o fundo do ecrã enquanto se está a jogar dependendo da hora do dia. Está implementado no rtc.c e rtc.h.



# Organização de código e estrutura

## Módulos

bitmap.c – 9%

Carregar, desenhar e apagar bitmaps.

*Desenvolvido por Henrique Ferrolho em:*

<http://difusal.blogspot.pt/2014/09/minixtutorial-8-loading-bmp-images.html>

cloud.c – 5,5%

Neste módulo as funções que são implementadas servem maioritariamente para controlar os casos dos sets de nuvens que aparecem no jogo.

Membro do grupo responsável: Maria Viana

game.c – 25%

Neste módulo é tratado de tudo um pouco: desde menus, as animações do jogo e como estas são tratadas com as colisões, implementados e desenhados o score e as vidas e a função principal do rtc, que é mudar a imagem de fundo conforme as horas.

Membro do grupo responsável: Diogo Yaguas/Maria Viana (50%/50%)

i8042.h/ i8254.h – 2%

Módulo apenas com macros.

jumpingPeanut.c – 17%

Funções que desenhavam o personagem e que eliminam o seu bitmap são implementadas neste módulo. É aqui que se encontra o grande ciclo de jogo que permite o uso dos três periféricos principais: o rato, o timer e o teclado.

Membro do grupo responsável: Diogo Yaguas/Maria Viana (50%/50%)

keyboard.c – 4%

Funções para uso do teclado, importado das aulas do laboratório 3.

Membro do grupo responsável: Diogo Yaguas

main.c – 4%

É neste módulo que se começa a execução do jogo: onde são chamadas todas as funções que fazem com que o jogo flua.

Membro do grupo responsável: Diogo Yaguas/Maria Viana (50%/50%)

mouse.c – 11%

Funções para uso do rato, importado das aulas do laboratório 4.

Membro do grupo responsável: Diogo Yaguas

rtc.c – 8%

Leitura da data e tempo atuais.

Membro do grupo responsável: Maria Viana

stateMachine.c – 2,5%

Módulo que contém o estado do menu do jogo tais como: Menu, Game, GameOver e Game\_Pause.

Membro do grupo responsável: Maria Viana

timer.c – 4%

Funções para manusear o timer, importado das aulas do laboratório 2.

Membro do grupo responsável: Diogo Yaguas

vbe.c – 4%

Importado das aulas do laboratório 5.

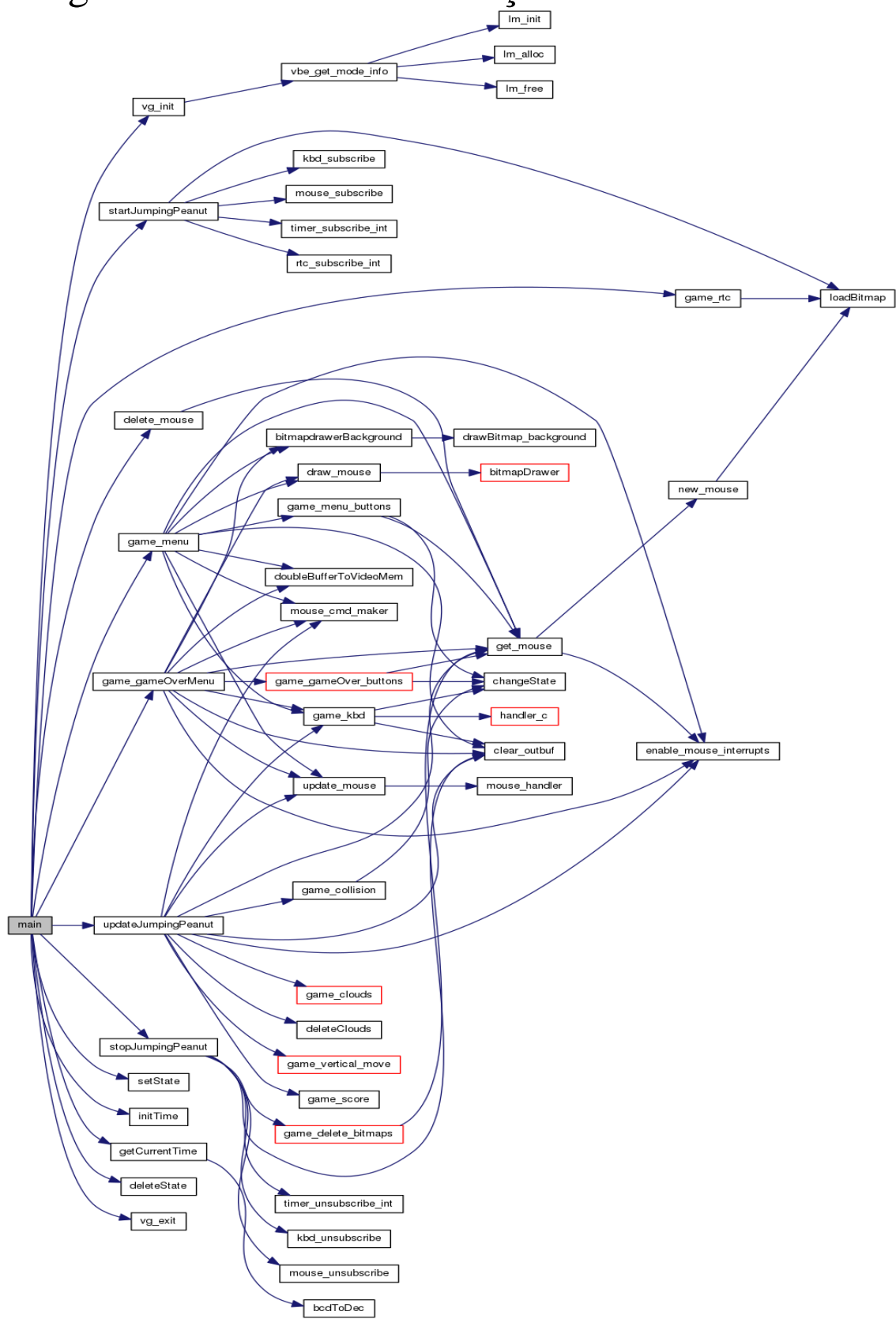
Membro do grupo responsável: Diogo Yaguas/Maria Viana (50%/50%)

video\_gr.c – 4%

Importado das aulas do laboratório 5.

Membro do grupo responsável: Diogo Yaguas/Maria Viana (50%/50%)

## Diagrama de chamada de funções



## Detalhes da implementação

Inicialmente iríamos replicar o tão conhecido jogo “Flappy Bird” apenas com uma única alteração: em vez de ser um pássaro, seria o Snoopy dos Peanuts. No entanto, ao longo da construção do jogo achamos por ventura fazer algo diferente da ideia inicial, então baseando-nos na mesma neste mesmo jogo substituímos os tubos por umas nuvens, dando mesmo a ideia de que o personagem estava a voar, e mudamos também as imagens de fundo do jogo e dos menus.

Decidimos implementar dois periféricos com que o jogador pudesse controlar a personagem da forma com se que sentisse mais confortável: com o rato ou com o teclado.

O nosso jogo foi implementado usando uma máquina de estados (stateMachine.c), que facilitou o nosso trabalho, tornando-o muito mais fluído e a construção do jogo por passos.

## Avaliação da unidade curricular

Na nossa opinião, apesar de as aulas teóricas serem bastante enriquecedoras, as aulas práticas já são o contrário pois temos que fazer tudo autonomamente sem quase ajuda nenhuma. O trabalho que desenvolvemos ao longo do semestre é bastante superior aos créditos que são atribuídos à disciplina.

Achamos também que os temas que não são abordados nos labs (neste caso RTC e UART) não deveriam limitar a nota do projeto.

## Instruções de Instalação

A instalação e execução do jogo devem ser feitas com acesso root ao sistema, efetuando o comando “su”. Seguidamente é necessário colocar o projeto na pasta proj e usar os seguintes scripts:

- sh install.sh
- sh compile.sh
- sh run.sh

O primeiro script serve para instalar o jogo e os outros para o executar.