

# *Space Invaders*

Laboratório de Computadores - 2015/2016

**LCOM1516-T6G12**

Bernardo Ferreira dos Santos Aroso Belchior - up201405381

Edgar de Lemos Passos - up201404131

4 de janeiro de 2016

# Space Invaders

Laboratório de Computadores - 2015/2016

## Conteúdo

1.	INSTRUÇÕES DE UTILIZAÇÃO.....	3
1.1.	MENU INICIAL .....	3
1.2.	SINGLEPLAYER.....	4
1.3.	MULTIPLAYER.....	6
1.4.	HIGHSCORES.....	7
	.....	7
1.5.	OPTIONS.....	8
1.6.	GAME OVER MENU.....	9
2.	ESTADO DO PROJETO .....	10
2.1.	DISPOSITIVOS UTILIZADOS.....	10
2.2.	TIMER.....	11
2.3.	TECLADO .....	12
2.4.	RATO .....	13
2.5.	PLACA GRÁFICA .....	14
2.6.	REAL TIME CLOCK .....	15
2.7.	SERIAL PORT.....	16
3.	ORGANIZAÇÃO E ESTRUTURA DO CÓDIGO .....	17
3.1.	ALIENS .....	17
3.2.	ANIMATION.....	17
3.3.	BITMAP.....	17
3.4.	BUTTON.....	17
3.5.	EVENTS.....	17
3.6.	FIFO.....	18
3.7.	FONT.....	18
3.8.	GAME OVER MENU.....	18
3.9.	HIGHSCORE.....	18
3.10.	I8042.....	18
3.11.	I8254.....	18
3.12.	KEYBOARD .....	18
3.13.	MENU.....	19
3.14.	MOUSE.....	19
3.15.	OPTIONS.....	19
3.16.	PLAYER.....	19
3.17.	PROJ .....	19
3.18.	PROJECTILE.....	19
3.19.	RTC.....	20
3.20.	SHIELD.....	20
3.21.	SINGLEPLAYER.....	20
3.22.	START MENU.....	20
3.23.	STATE.....	20

3.24.	<i>TIMER</i> .....	20
3.25.	<i>UART</i> .....	21
3.26.	<i>VBE</i> .....	21
3.27.	<i>VERSUS MULTIPLAYER</i> .....	21
3.28.	<i>VIDEO GR</i> .....	21
4.	DETALHES DA IMPLEMENTAÇÃO .....	23
5.	CONCLUSÕES.....	24
	APÊNDICE.....	25

## 1. Instruções de utilização

### 1.1. Menu inicial

Menu inicial do projeto, que contém os submenus principais do jogo:

- *Singleplayer*;
- *Multiplayer*;
- *Highscores*;
- *Options*.

A opção *Exit* permite sair do serviço *proj* e voltar ao ambiente normal de consola do *Minix*.

O menu permite o uso do rato para clicar nos diferentes botões, ou então as teclas para cima/baixo para seleccionar as diferentes opções, usando a tecla “Enter” para executar a ação subjacente à opção escolhido. Ao pressionar a tecla “Esc”, nenhum botão será selecionado.



FIGURA 1 - MENU INICIAL

## 1.2. Singleplayer



FIGURA 2 - *SINGLEPLAYER*

Acima presenciamos o modo de um só jogador.

O objetivo do jogo é exterminar todos os tipos de extraterrestres presentes na figura 2, estes são as imagens a branco no centro do ecrã. Caso o jogador não o consiga concretizar antes dos invasores atingirem os escudos, o jogador perde. Outro critério de derrota é o número de vidas, representadas no canto inferior direito. O utilizador tem, por defeito, três vidas, e perde uma cada vez que é atingido. Quando o jogador tiver apenas uma vida e chocar contra um projétil, perde o jogo.

Quando morto, cada tipo de extraterrestre fornece uma quantia diferente de pontos ao utilizador, contabilizada no canto inferior esquerdo do ecrã (*score*), que é a medida de comparação entre diferentes jogos.

O jogador também tem a seu dispor um conjunto de quatro escudos que pode utilizar para se proteger dos projéteis lançados pelos invasores.

No topo do ecrã, aleatoriamente, aparecerá uma nave a vermelho que fornece pontos extra quando é atingida.

É também possível sair do jogo pressionando a tecla “Esc”.

Saindo do modo *singleplayer* obrigará o programa a decidir se a pontuação do jogador é alta o suficiente para entrar na tabela das melhores pontuações. Caso o resultado seja afirmativo, será apresentado ao jogador o “Game Over Menu” (ver pág. 9), caso contrário, este será levado para o menu inicial.

### 1.3. Multiplayer



FIGURA 3 - *MULTIPLAYER*

Acima apresenta-se o modo multijogador que consiste em dois jogadores que disparam projéteis um contra um outro. Quando um deles é atingido, a pontuação do adversário é incrementada. Ao fim do tempo mostrado no canto superior direito é apurado o vencedor. Este será o que tiver mais pontos. Esta pontuação não será adicionada à tabela das melhores pontuações.

#### 1.4. Highscores

Do lado direito está representado do menu das melhores pontuações com o nome (3 letras, ver “Game Over Menu”), data (dia/mês/ano) e pontuação respectivas.

O menu permite o uso do cursor para clicar no botão “Back” para voltar ao menu inicial ou o uso da tecla “Esc”.

HIGHSCORES		
NAME	DATE	SCORE
AAA	2/1/16	99
AAA	2/1/16	99
AAA	2/1/16	99
AAA	2/1/16	99
AAA	2/1/16	99
AAA	2/1/16	99
AAA	2/1/16	99
XAA	2/1/16	89
YAA	3/1/16	79
AAA	4/1/16	89
BACK		

FIGURA 4 - *HIGHSCORES*



### 1.5. Options

Abaixo está representado o menu de opções em que é possível escolher o tipo de controlador a utilizar no modo de um jogador. As opções são: *keyboard*, que permite usar as teclas para a direita e esquerda e barra de espaços do teclado para movimentar a nave do jogador; e *mouse* que utiliza o movimento do rato e o botão do lado esquerdo para mover e disparar, respetivamente.



O menu é passível de ser navegado utilizando as teclas para a direita e esquerda para alterar o controlador a usar, assim como as teclas “Esc” e “Enter” para cancelar e aceitar as alterações feitas. É também possível utilizar o rato para estas ações.

FIGURA 5 - *OPTIONS*

### 1.6. Game Over Menu

Caso a pontuação do jogador no modo *singleplayer* seja suficiente para figurar na tabela das melhores pontuações, este será apresentado com o seguinte menu.



FIGURA 6 - *GAME OVER MENU*

O menu de fim de jogo permite ao utilizador introduzir um nome identificativo de três letras que, mais tarde, será adicionado à tabela das melhores pontuações.

Este menu poderá ser navegado utilizando o rato.

## 2. Estado do projeto

### 2.1. Dispositivos utilizados

<i>Dispositivo</i>	<i>Utilização</i>	<i>Interrupção?</i>
<i>Timer</i>	Atualizações do estado do jogo	Sim
<i>Teclado</i>	Interface do jogo com o utilizador e navegação nos menus	Sim
<i>Rato</i>	Interface do jogo com o utilizador e navegação nos menus	Sim
Placa Gráfica (modo vídeo)	Desenho de imagens	Não
<i>Real Time Clock</i>	Obtenção de data das melhores pontuações	Não
<i>Serial Port</i>	Comunicação entre dois computador para o modo multijogador	Sim

### 2.2. *Timer*

O *timer* presente no *Minix* é utilizado no sentido de atualizar o estado do programa. Quer seja necessário desenhar imagens, mover extraterrestres e disparar projéteis. Praticamente todo o tipo de informação é modificado pelas interrupções do temporizador, sendo este indispensável para o bom funcionamento de todo o projeto.

### 2.3. Teclado

O teclado, sendo um dispositivo não tão importante para o sucesso do trabalho, também tem a sua fatia de atenção. Os jogadores quer no modo de apenas um jogador, quer em multijogador têm a opção de utilizar as teclas para controlar as suas naves. Contudo, não é só nos modos de jogo que este dispositivo é usado. Na interação jogador-menu, o teclado tem um papel alternativo, mas simultâneo, ao do rato, alargando as opções de escolha para o utilizador.

## 2.4. Rato

Este periférico mantém uma relação chegada ao utilizador, uma vez que é o modo mais utilizado de navegação nos menus. No entanto, também poderá ser utilizado no modo de um jogador para movimentar a nave e lançar projéteis.

## 2.5. Placa Gráfica

A placa gráfica é um componente chave do projeto, dado que permite desenhar imagens criadas pelo grupo, utilizando o modo **RGB (5:6:5)** com 1024 pixéis de largura por 768 de altura.

O grupo decidiu utilizar a técnica de *double buffering* de modo a fornecer uma experiência mais suave ao jogador.

De modo a detetar colisões entre entidades do jogo, o grupo desenvolveu um algoritmo de deteção de colisões entre retângulos.

Numa tentativa de embelezar o ambiente de jogo, foram criadas animações utilizando *bitmaps*.

Para ser possível o envio de informação textual, o grupo criou um algoritmo para escrever inteiros, caracteres e *strings* através de um *bitmap* baseado na tabela **ASCII**.

## 2.6. *Real Time Clock*

O relógio em tempo real é utilizado no projeto com o intuito de ler a data de obtenção de uma das melhores pontuações. Como tal, o relógio só é acedido quando necessário, não sendo necessário gerar interrupções.



### 2.7. *Serial Port*

Parcialmente implementada nos módulos FIFO e UART. O grupo não conseguiu desenvolver a infraestrutura necessária para o bom funcionamento da porta de série a tempo da entrega.

## 3. Organização e estrutura do código

### 3.1. Aliens

Lida com todo o tipo de interações jogador-extraterrestres.

Este módulo foi desenvolvido maioritariamente pelo elemento Edgar Passos (80%).

### 3.2. Animation

Contém as *sprite* de cada animação, permite criar animações a partir de nomes de ficheiros, ou de *bitmaps* já carregados.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.3. Bitmap

Carrega, desenha e apaga *bitmaps*. Código publicado por Henrique Ferrolho ([origem](#)), no seu [blog](#), com variadas modificações, nomeadamente em termos de carregamento de imagens. As imagens em formato *bmp* são guardadas pelo programa de edição de imagens “Gimp 2” com as linhas invertidas verticalmente. Enquanto o código original lê a imagem como está e a desenha invertida, o código alterado ao carregar a imagem já a espelha segundo um eixo horizontal, e portanto, encontra-se em memória como deverá ser impressa.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.4. Button

Cria, lida com colisões, desenha e destrói botões. É permitido criar botões com *bitmaps* como fundo e/ou texto.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.5. Events

Lida com os eventos lançados pelas interrupções, quer estes sejam cliques do rato, interrupções do *timer* ou interação do utilizador do teclado.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.6. FIFO

Contém a implementação de uma fila em C, para utilização com a *serial port*.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### 3.7. Font

A partir de um *bitmap*, com um formato específico, que contém caracteres ordenados pelo seu código ASCII, desenha inteiros, caracteres e *strings*.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.8. Game Over Menu

Menu criado quando um jogador finaliza o jogo de único jogador e a sua pontuação é suficiente para entrar na tabela das melhores pontuações.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.9. Highscore

“Classe” que contém a informação das dez melhores pontuações obtidas no modo de um jogador. Lê e escreve para um ficheiro “highscore.txt”.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### 3.10. 18042

Importado do código das aulas práticas do laboratório 3.

### 3.11. 18254

Importado do código das aulas práticas do laboratório 2.

### 3.12. Keyboard

Importado do código das aulas práticas do laboratório 3.

### **3.13. Menu**

Contém todos os botões pertencentes a cada menu e trata do seu desenho e colisão.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### **3.14. Mouse**

Importado do código das aulas práticas do laboratório 4.

### **3.15. Options**

Menu de opções. Contém todas as interações do utilizador com o menu.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### **3.16. Player**

Contém toda a informação sobre um jogador e trata da sua colisão com projéteis.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### **3.17. Proj**

Módulo principal: inicializa variáveis, subscreve interrupções e inicializa o modo gráfico. Contém o ciclo principal do jogo. Após isso, apaga as variáveis, sai do modo gráfico e faz *unsubscribe* às interrupções subscritas.

Este módulo foi desenvolvido igualmente pelos elementos do grupo (50%/50%).

### **3.18. Projectile**

Trata da criação, desenho e destruição dos projéteis. Implementa uma lista duplamente ligada.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### **3.19. RTC**

Lê informação contida nos registos do *Real Time Clock*.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### **3.20. Shield**

Contém toda a informação relativa aos escudos do jogo.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### **3.21. Singleplayer**

Trata todos os acontecimentos relativos ao modo de um só jogador.

Este módulo foi desenvolvido igualmente pelos elementos do grupo (50%/50%).

### **3.22. Start Menu**

Menu inicial e tudo o que lhe está subjacente.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### **3.23. State**

Contém todos os estados e lida com as suas transições, eliminado o estado anterior, antes de evoluir para o seguinte.

Este módulo foi desenvolvido unicamente pelo elemento Bernardo Belchior.

### **3.24. Timer**

Importado do código das aulas práticas do laboratório 2.

### 3.25. UART

Módulo contendo o código relativo à porta de série.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### 3.26. VBE

Importado do código das aulas práticas do laboratório 5.

### 3.27. *Versus Multiplayer*

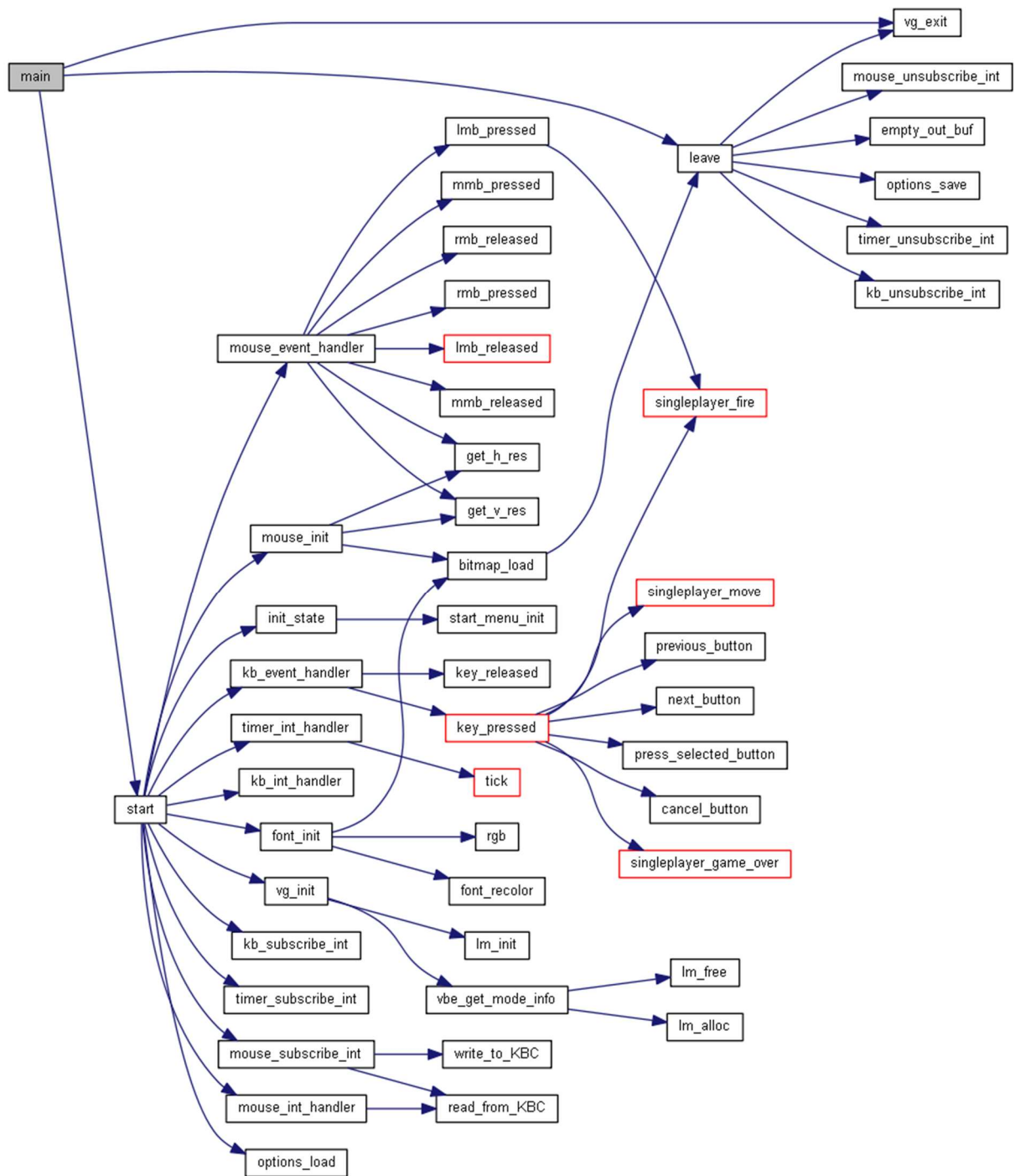
Trata todos os acontecimentos relativos ao modo multijogador.

Este módulo foi desenvolvido unicamente pelo elemento Edgar Passos.

### 3.28. *Video Gr*

Importado do código das aulas práticas do laboratório 5.

### 3.29. Call Graph



## 4. Detalhes da implementação

O grupo utilizou código por camadas de modo a facilitar a utilização e escrita de funções mais complexas. É exemplo o módulo “font”, que permite escrever texto de forma muito simples, apesar da complexidade que se encontra por detrás do código. A facilidade de criar menus e botões é mais uma prova da boa estrutura construída pelo grupo.

O módulo *bitmap*, apesar de ter sido inicialmente copiado (ver módulo *Bitmap*) foi muito modificado de modo a adaptar-se às necessidades do grupo.



## 5. Conclusões

O grupo sente que se esforçou e dedicou muitas horas a este projeto, contudo, a falta de informação em certos aspetos são menos oportunos.

O elemento Bernardo Belchior utilizou muito do seu tempo para a pesquisa acerca da chamada de funções em *assembly* a partir de código em C. Por sua vez, o elemento Edgar Passos denotou muita dificuldade na implementação da porta de série que acabou por não ser finalizada.

## Apêndice

Devido à utilização de imagens e da necessidade de utilizador caminhos consistentes, o programa inclui algumas diferenças em relação ao convencional *make*.

Instruções de instalação:

- Entrar no diretório “scripts”
- Correr o *script* “install.sh”
- Correr o *script* “compile.sh”
- Correr o *script* “run.sh”