

**MIEIC – Dezembro 2017**

**Projeto Final**

**LCOM**

**Turma 7 - Grupo 9**

**Bruno Alexandre Oliveira Dias – up201504859**

**Maria Eduarda Santos Cunha – up201506524**

Índice

[1. Introdução 3](#_Toc502432181)

[2. Obstacle Course 3](#_Toc502432182)

[3. Instruções de Utilização 3](#_Toc502432183)

[4. Estado do Projeto 3](#_Toc502432184)

[4.1. Timer 3](#_Toc502432185)

[4.2. Teclado 4](#_Toc502432186)

[4.3. Placa de Vídeo 4](#_Toc502432187)

[4.4. Rato 4](#_Toc502432188)

[4.5. RTC 4](#_Toc502432189)

[5. Estrutura e Organização do Código 4](#_Toc502432190)

[5.1. main.c 4](#_Toc502432191)

[5.2. obstacleCourse.h 4](#_Toc502432192)

[5.3. obstacleCourse.c 4](#_Toc502432193)

[5.4. mouse.h e mouse.c 5](#_Toc502432194)

[5.5. keyboard.h e keyboard.c 5](#_Toc502432195)

[5.6. timer.h e timer.c 5](#_Toc502432196)

[5.7. graphics.h e graphics.c 5](#_Toc502432197)

[5.8. vídeo\_gr.h e vídeo\_gr.c 5](#_Toc502432198)

[5.9. bitmap.h e bitmap.c 6](#_Toc502432199)

[5.10. utilities.h 6](#_Toc502432200)

[5.11. i8042.h 6](#_Toc502432201)

[6. Detalhes de Implementação 6](#_Toc502432202)

[7. Conclusões 7](#_Toc502432203)

[8. Apêndice: Instruções de Instalação 7](#_Toc502432204)

# Introdução

O presente relatório serve de complemento ao projeto final desenvolvido na cadeira de Laboratório de Computadores, de forma a explicar e analisar o código desenvolvido de forma mais extensa e específica.

Pensamos que realizamos os requerimentos esperados.

# Obstacle Course

O jogo desenvolvido, Obstacle Course, tem como objetivo principal o desvio de obstáculos por parte do jogador.

Existem 3 faixas sobre as quais o jogador pode deslocar o seu carro, os obstáculos que se tentam evitar e moedas, que, se apanhadas, acumulam pontos.

# Instruções de Utilização

As instruções de instalação encontram-se no campo 8. Apêndice: Instruções de Instalação.

O menu principal possui as opções *Play* para jogar, *Score* para consultar as pontuações e *Exit* para sair da aplicação.

Quando no modo de jogo, o utilizador deve tentar desviar o carro dos obstáculos que fazem perder vidas e apanhar as moedas que permitem acumular pontos.

O jogador possui 5 vidas, portanto embater em 5 obstáculos termina o jogo.

De forma a deslocar o carro, o utilizador pode optar por usar as teclas w e s, para andar para cima e para baixo respetivamente, ou recorrer ao rato. A deslocação por via do teclado permite um deslocamento rápido entre as 3 faixas, ao passo que com o rato, de acordo com os seus movimentos no sentido positivo ou negativo (segundo o eixo dos y), o carro desloca-se lentamente podendo ocupar posições intermédias.

No menu principal, pode-se pressionar quer a tecla *esc* ou o botão *Exit* para voltar à consola.

No modo de jogo, pressionar *esc* leva de volta ao menu principal.



Figura 2 Jogo com carro deslocado por meio das teclas w e s

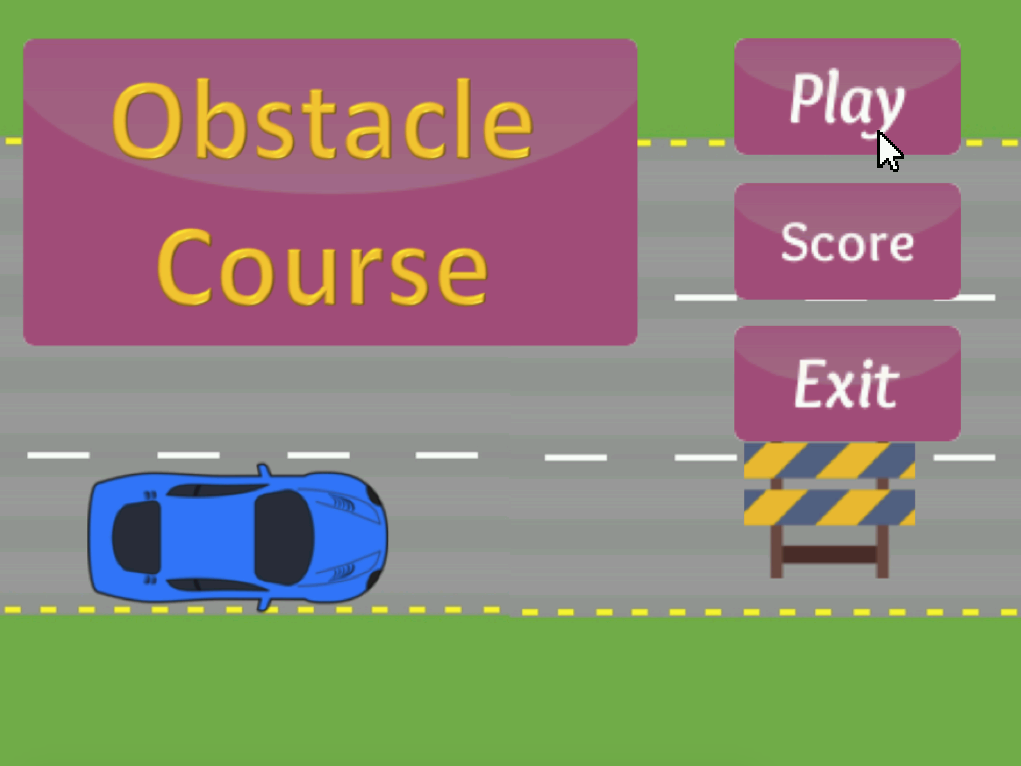


Figura Menu principal com opção Play selecionada pelo rato

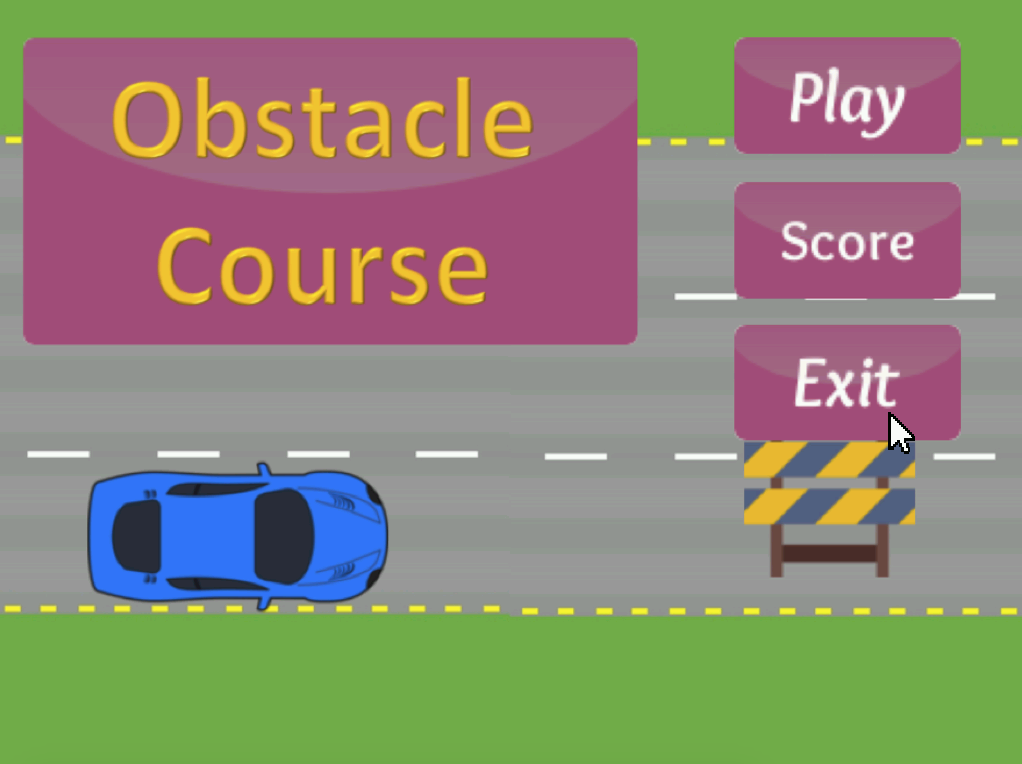


Figura 6 Menu principal com opção Exit selecionada pelo rato

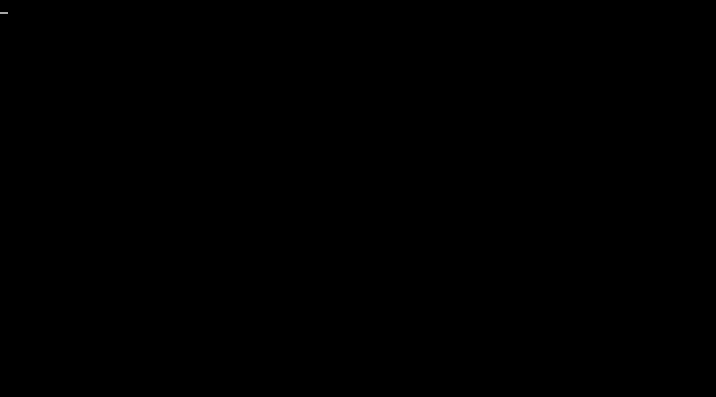


Figura 7 Consola após selecionar a opção Exit

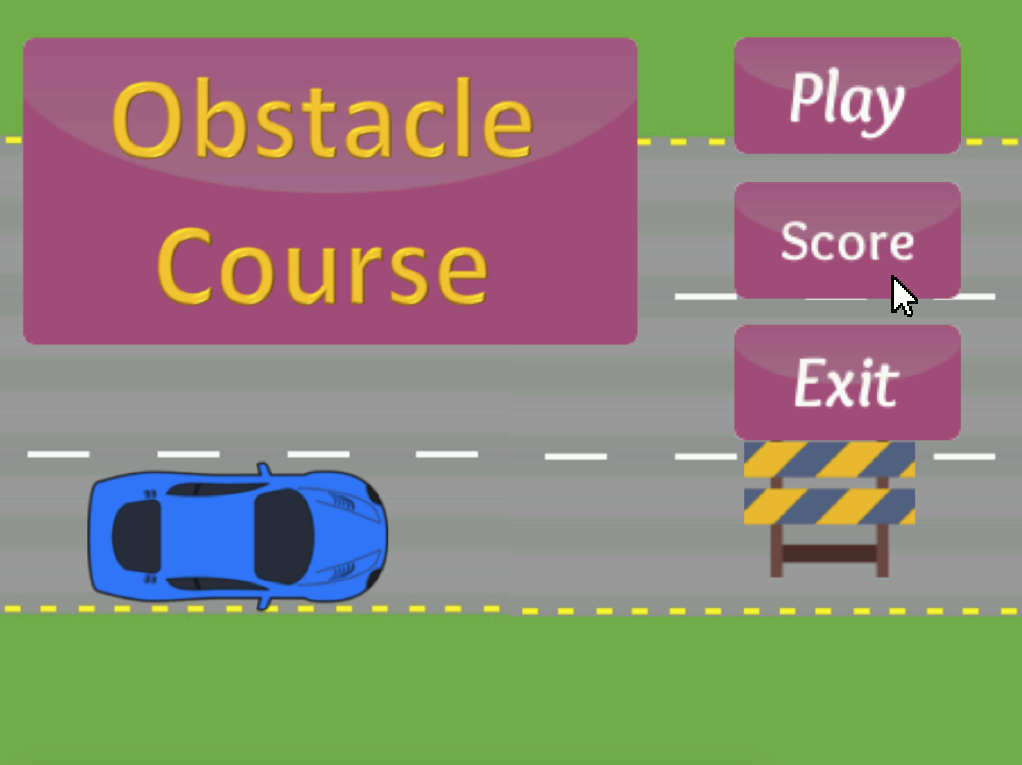


Figura 4 Menu principal com opção Score selecionada pelo rato



Figura 3 Jogo com carro deslocado por meio do rato

# Estado do Projeto

O projeto encontra-se implementado conforme previsto na especificação do projeto, à exceção de:

* Foram acrescentados alguns obstáculos;
* Foi mudado o sistema de pontuação;
* Não chegamos a implementar a ligação porta série, que constituía um extra que gostávamos de ter aplicado, de forma a tornar o jogo *multiplayer*.

Assim, atualmente, o carro desloca-se para cima e para baixo por ação do teclado ou do rato, conforme preferência do utilizador. Caso embata num obstáculo, perde vidas e, numa moeda, acumula pontos. Estes elementos de débito e soma de pontos são gerados aleatoriamente nas faixas ao longo do tempo.

|  |  |
| --- | --- |
| Especificação do Projeto | Versão Final |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Dispositivo I/O** | **Utilização** | **Interrupção?** |
| Timer | Atualização do estado de jogo | Sim |
| Teclado | Movimentar carro | Sim |
| Placa de Vídeo | Exibir ecrã de jogo e menus | Não |
| Rato | Selecionar opções nos menus e movimentar carro | Sim |
| RTC | Guardar registos das pontuações | Não |

## Timer

Descrever funcionalidade, referindo as funções onde é usado.

## Teclado

Descrever funcionalidade, referindo as funções onde é usado.

É utilizado no controlo do jogo para deslocar o carro ao longo das faixas, segundo as teclas w e s, para cima e para baixo, respetivamente.

## Placa de Vídeo

Descrever funcionalidade, referindo as funções onde é usado.

Indicar modo de vídeo, resolução, modo de cor e número de cores.

Indicar se usa Double/triple buffering.

Indicar se tem objetos com movimento (deteção de colisão, sprites animadas).

Indicar se usa fonts.

Indicar funções VBE.

## Rato

Descrever funcionalidade, referindo as funções onde é usado.

É utilizado para optar entre botões, *Start* para iniciar o jogo, *Score* para consultar as pontuações e *Exit* para sair da aplicação.

Ainda, no modo de jogo, de acordo com a sua deslocação (sentido positivo ou negativo segundo o eixo dos y), permite movimentar o carro ao longo da pista.

## RTC

Descrever funcionalidade, referindo as funções onde é usado.

Indicar se é usado para ler data/tempo.

Indicar se é usado para gerar um alarme, interrupções periódicas.

# Estrutura e Organização do Código

## main.c

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias e Eduarda Cunha

## obstacleCourse.h

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias e Eduarda Cunha

## obstacleCourse.c

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias e Eduarda Cunha

## mouse.h e mouse.c

Código importado da submissão do laboratório 4 das aulas práticas, com pequenas alterações de forma a corresponder às necessidades do projeto.

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias

## keyboard.h e keyboard.c

Código importado da submissão do laboratório 3 das aulas práticas, com pequenas alterações de forma a corresponder às necessidades do projeto.

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias

## timer.h e timer.c

Código importado da submissão do laboratório 2 das aulas práticas, com pequenas alterações de forma a corresponder às necessidades do projeto.

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Eduarda

## graphics.h e graphics.c

Código importado da submissão do laboratório 5 das aulas práticas, com pequenas alterações de forma a corresponder às necessidades do projeto.

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Pessoa

## vídeo\_gr.h e vídeo\_gr.c

Código importado da submissão do laboratório 5 das aulas práticas, com pequenas alterações de forma a corresponder às necessidades do projeto.

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Bruno Dias

## bitmap.h e bitmap.c

Que mudanças tivemos de fazer?

Estes ficheiros são maioritariamente da autoria do Henrique Ferrolho e encontram-se disponibilizados em <http://difusal.blogspot.pt/2014/09/minixtutorial-8-loading-bmp-images.html>, sendo que sofreram algumas alterações a nível de \_\_.

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Pessoa

## utilities.h

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Pessoa

## i8042.h

É da nossa autoria? Que mudanças tivemos de fazer? Url da fonte

Parágrafo a descrever código do módulo.

Descrição das estruturas principais.

Dizer quem fez o quê.

Peso no total do projeto: \_%

Responsável: Pessoa

Gráfico de Invocação de Funções

It can be generated automatically by Doxygen

May want to limit the depth so that the graph generated can be included in a figure in the report. In this case, include the full Doxygen documentation in the SVN repository

Include also a short description of the main functions

These must include functions that call driver\_receive()

# Detalhes de Implementação

This is where you can show your domain of the course’s topics. In general, you should speak about topics:

That were covered in the lectures, but that required some ingenuity in their application to your project (e.g. layering, event driven code, state machines, object orientation, frame generation, assembly code, ...)

Details regarding the use of the RTC and the UART are also important

That were not covered in the lectures/labs and that you had to learn by yourself (and may be you wished we had talked about it) (e.g. collision detection, call of assembly functions from C, ...)

# Conclusões

De forma geral, pensamos que tivemos um bom desempenho na cadeira ao longo do semestre.

Ambos nos encontramos a repetir a cadeira, pelo que temos uma perspetiva bem diferente da que possuíamos o ano passado, mas algumas ideias mantêm-se. Continuamos a ser da opinião de que a cadeira se insere na posição temporal errada do curso, visto que envolve a aprendizagem de muitos conhecimentos novos que ainda não foram lecionados ou suficientemente aprofundados nesta fase.

Passamos a explicar melhor a perspetiva do ano passado: é um verdadeiro desafio lidar com a pressão temporal de ter entregas quinzenais pela primeira vez, ter de programar em C quando recebemos apenas algumas luzes em PROG, trabalhar numa consola começando por usar comandos um pouco sem saber para que servem etc.

Este ano mantemos a opinião de que, para o aluno do 2º ano, a cadeira continua a reunir as dificuldades apresentadas, mas, enquanto alunos do 3º ano, a dificuldade da linguagem de programação e de trabalhar com a consola desapareceram. Assim, restou-nos apenas a gestão de tempo para trabalhar, dado que estávamos a realizar 6 cadeiras cada um e o trabalho em conjunto era dificultado por não sermos do Porto e de cidades diferentes.

Cumprimos a maior parte dos objetivos a que nos tínhamos proposto na especificação do projeto, tendo deixado apenas por fazer a ligação porta série, que considerávamos um extra ao nosso alcance, mas, dado que tínhamos mais 3 entregas das unidades curriculares do 3º ano para as 2 semanas de intervalo letivo, tornou-se impossível. Ainda, realizamos algumas alterações ao inicialmente pensado, quer para incluir componentes que, caso contrário, por falta de tempo, não teríamos conseguido implementar (troca entre uso do RTC para fazer um registo das pontuações para display das horas e data no fim do jogo), quer para acrescentar funcionalidades (inclusão de mais obstáculos e sistema de vidas).

# Apêndice: Instruções de Instalação

No terminal, invocar os seguintes comandos dentro da pasta do projeto:

**sh install.sh** se estiver a correr pela primeira vez ou ocorrer mudança dos recursos (imagens) a utilizar;

**sh compile.sh** para compilar todos os ficheiros necessários;

**sh run.sh** para correr o programa desenvolvido e jogar.