



FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Stick Hero

Laboratório de Computadores
Turma 6, Grupo 10



Jorge Miguel Rodrigues Ferreira (up201207133)
up201207133@fe.up.pt

Nuno Filipe Sousa e Silva (up201404380)
up201404380@fe.up.pt

Índice

Introdução	3
Instalação	3
Instruções de utilização do jogo	4
Menu Inicial	4
New Game	5
Perder	6
Max Score	7
Clock	8
Exit	8
Estado do Projecto	9
Periféricos	9
Descrição dos módulos	11
Divisão de trabalho	12
Diagrama de chamadas	13
Detalhes da Implementação	14
Conclusão	16
Auto-Avaliação	17
Bibliografia	18

Introdução

Este projeto visa mostrar a interação entre diferentes periféricos em ambiente minix através da criação do jogo “Stick Hero”.

“Stick Hero” é um jogo onde o jogador incorpora a posição de um ninja que tem como objetivo fazer crescer uma barra até que tenha um tamanho que chegue e que não ultrapasse a plataforma à qual está a tentar alcançar.

O jogo usa os seguintes periféricos: teclado, timer, rato, RTC e a placa de vídeo em modo gráfico.

Foi usada linguagem C e o IDE Eclipse no desenvolvimento do projeto.

Instalação

Após adquirir os ficheiros do projeto com o comando:

svn checkout <https://svn.fe.up.pt/repos/lcom1516-t6g10/>

Entrar na pasta source com:

cd lcom1516-t6g10/proj/StickHero/source/

Elevar os privilégios com:

su

Copiar o ficheiro para /etc/system.conf.d/ com:

sh cpyconf.sh

E para correr agora é só preciso fazer:

make

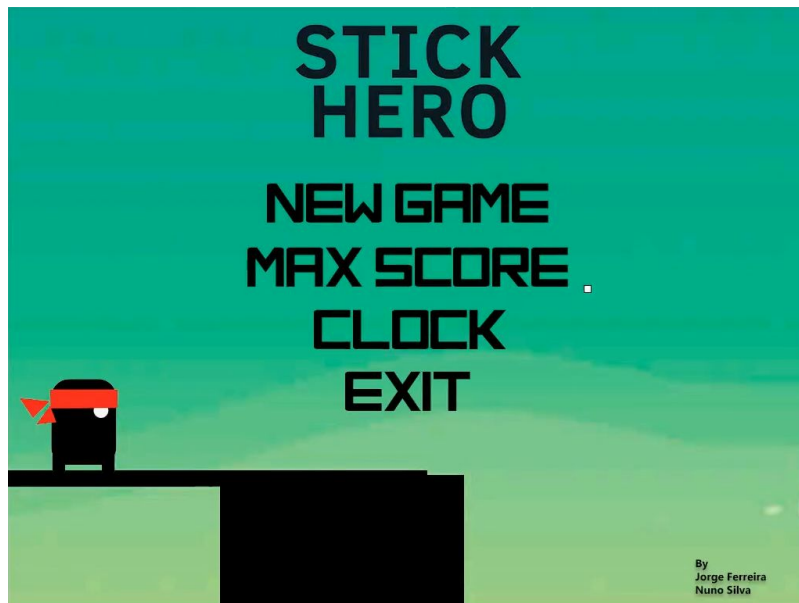
E iniciar o jogo com:

sh run.sh

Instruções de utilização do jogo

Pequeno vídeo que demonstra o jogo e as suas funcionalidades: [Vídeo](#).

Menu Inicial

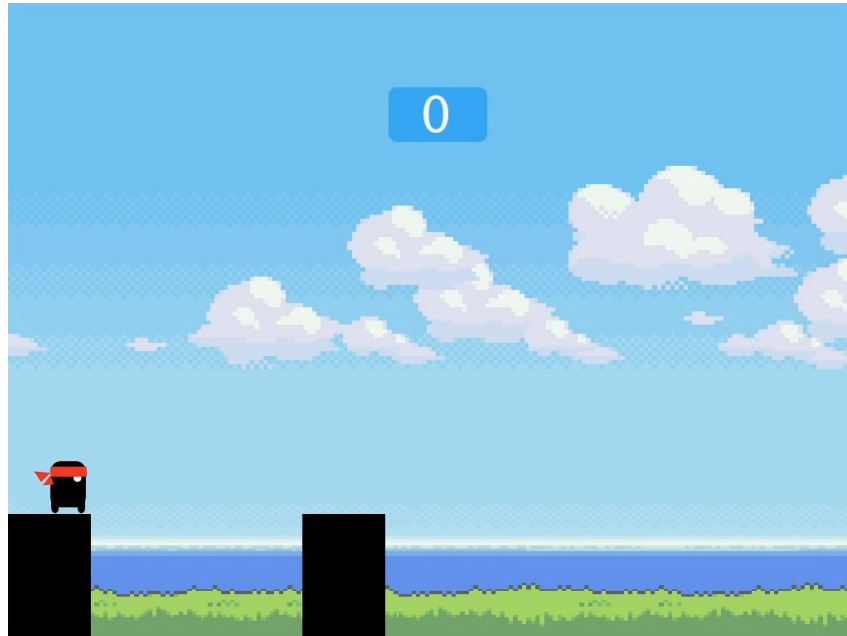


O rato é usado para movimentar o ponteiro e o botão esquerdo do rato para selecionar uma das quatro opções.

(clique nos nomes das opções que estão a azul e sublinhadas para ir para a descrição da opção)

- [New Game](#)
- [Max Score](#)
- [Clock](#)
- [Exit](#)

New Game



Aqui é onde se pode jogar o jogo.

É mostrado ao jogador uma imagem semelhante à de cima, com o ninja em cima de uma plataforma e uma plataforma, que foi gerada aleatoriamente, à frente do ninja.

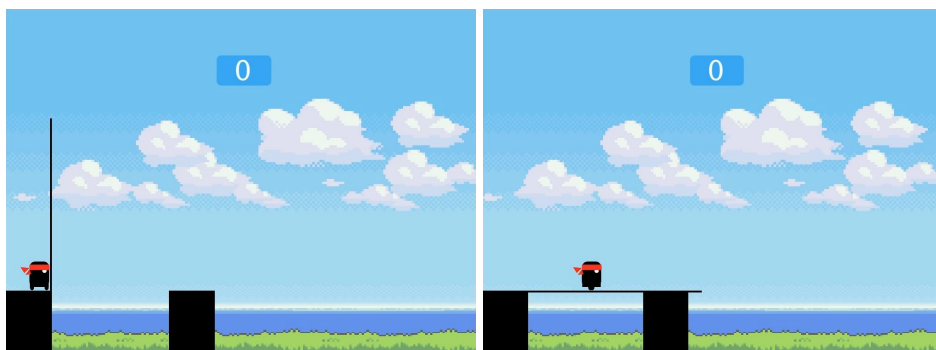
O objetivo do jogo é criar uma barra onde a barra, ao cair, chegue à plataforma sem a ultrapassar.

O jogador pode sair do jogo carregando na tecla “**ESC**” o que o leva ao estado de [Perder](#).

Para criar a barra, o jogador carrega na tecla “**Espaço**” e enquanto não largar a tecla “**Espaço**” a barra vai aumentando de tamanho.

Quando se larga a tecla “**Espaço**” a barra vai cair e o ninja vai percorrer a barra até à próxima plataforma.

As duas imagens abaixo mostram a barra quando se prime a tecla “**Espaço**” e quando se larga a tecla “**Espaço**”.

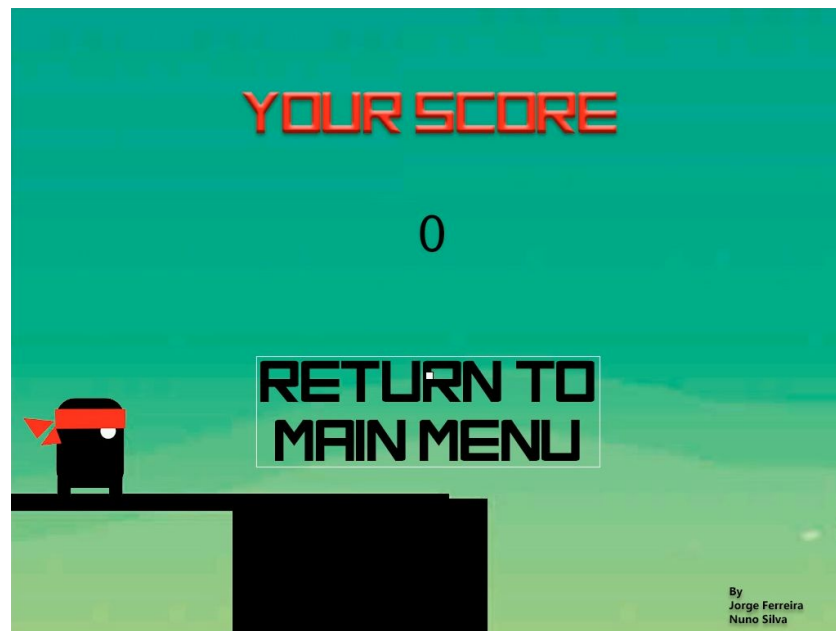


Se a barra for curta demais ou grande demais, o ninja cai e o jogador é levado ao estado de [Perder](#).

Se a barra tiver o tamanho certo, a plataforma onde o ninja se encontra após percorrer a barra vai ser deslocada para a posição inicial e será gerada uma nova plataforma que será deslocada do lado direito do ecrã até a uma posição também gerada aleatoriamente.

O jogo tem varias animações na qual é possível observar no [Vídeo](#).

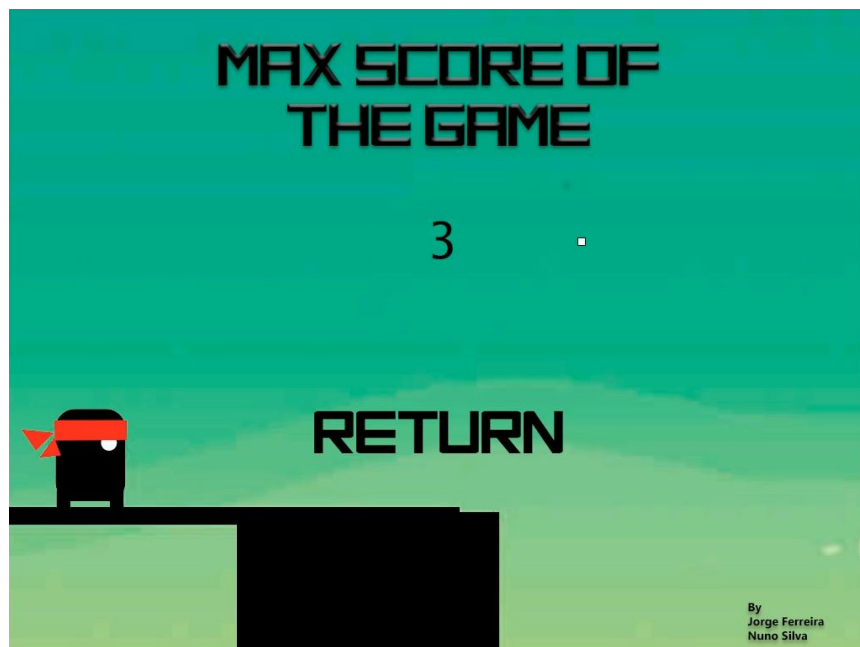
Perder



Aqui é mostrada a pontuação que o jogador fez na jogada realizada.

Ao clicar com o botão do lado esquerdo do rato em **“Return to Main Menu”** o jogador volta ao [Menu Inicial](#).

Max Score

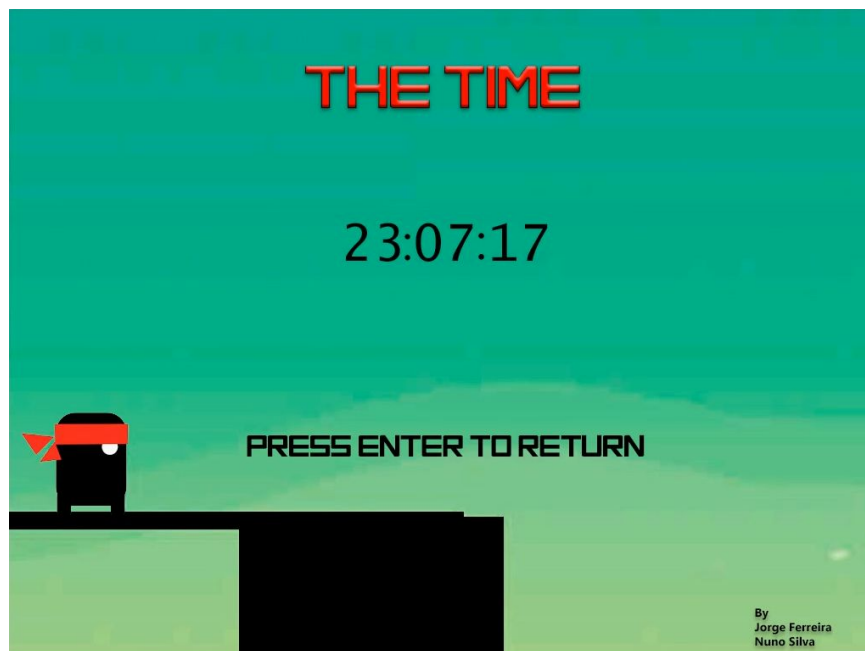


Aqui é mostrada a pontuação máxima registada no jogo.

A pontuação máxima é sempre atualizada de forma a que neste menu seja sempre mostrada a pontuação máxima de todas as sessões de jogos realizadas.

Quando se clicar com o botão do lado esquerdo do rato em **“Return”** o utilizador volta ao [Menu Inicial](#).

Clock



Nesta opção do menu inicial é apresentado no ecrã um relógio com a hora atual que é atualizada consoante o tempo passa.

Ao premir no teclado a tecla “**Enter**” o utilizador volta ao [Menu Inicial](#).

Exit



Se o utilizador do jogo clicar em “**Exit**” o jogo irá encerrar-se surgindo a janela anterior do minix já em modo de texto.

Estado do Projecto

Dispositivo	Modo de uso	Interrupções
Timer	Controlar os frames	Sim
KBD	Retorno na secção do relógio e uso para criar a barra no jogo	Sim
Mouse	Navegação no menu e seleção das opções	Sim
Video Card	Mostrar no ecrã os frames processadas	Não
RTC	Mostrar as horas atuais num menu específico	Não

Periféricos

Rato

Foi usado para escolher opções, sendo que as coordenadas do rato foram usadas para determinar a posição onde o rato deveria de estar e caso o rato estivesse por cima de uma das opções e fosse pressionado o botão esquerdo do rato, o utilizador entrava nessa opção. (Menu.c l.86 a l.127, l.183 a l.194 e menufinal.c l.54 a l.77).

Placa Gráfica

A placa gráfica foi utilizada basicamente para mostrar no ecrã imagens que foram processadas.

Foi utilizada a resolução 1024x768 (0x117) onde cada pixel apresenta 16 bits que representam 5 bits para a cor vermelha, 6 bits para a cor verde e 5 bits para a cor azul. No total existem 65 536 cores (Utilidades.c l.5).

Para utilizar as imagens no minix, foi utilizada uma função externa permitida pelo professor para leitura de imagens no formato BMP (BMP.c l.4 a l.98).

Para apagar da memória a alocação feita pela função anterior, foi realizada uma função para libertar essa memória (BMP.c l.100 a l.104).

Alguns aspectos importantes utilizados foram:

- Duplo Buffer para minimizar o chamado efeito “snow”. Em vez de escrever diretamente na memória da gráfica é primeiro escrito para um buffer no qual após o frame estiver totalmente processado, esse buffer será copiado para a memória da gráfica (Graficos.c l.256 a l.295);
- Movimentação de objetos onde existe a movimentação do rato e do ninja por exemplo (Graficos.c l.441 a l.453)
- Utilização de imagens com números para a apresentação das horas e das pontuações tanto no jogo como nos menus. (Graficos.c l.711 a l.754).

Teclado

O teclado foi usado principalmente no jogo. A ação principal do jogo é feita ao premir a tecla “Espaço”. Enquanto o jogador tiver a tecla “Espaço” pressionada a barra vai crescendo até o mesmo largar a tecla. Caso o jogador esteja a jogar e queira sair para o menu, ele só precisa de carregar na tecla “Esc”.

Na opção clock, para sair para o menu o utilizador só precisa de carregar na tecla “Enter”. (Jogo.c l.200 a l.205, l.222 a l.234, l.274 e l.275, l.283 a l.295, Menu.c l.62 a l.64, l.170 a l.172, l.223, l.259 a l.261, menufinal.c l.51 e l.52).

Timer

O timer foi usado praticamente para controlar os frames do jogo. É desenhado um frame a cada interrupção do timer, o que significa que o jogo corre a 60 fps (Menu.c l.66 a l.84, l.173 a l.181, l.263 a l.264 - Jogo.c l.207 a l.237, l.277 a l.298).

RTC

O “real time clock” foi utilizado para que a qualquer momento que o utilizador queira verificar as horas, o possa fazer no próprio jogo clicando na opção do menu inicial “clock”. Este é atualizado a cada ciclo. Como não foram utilizadas interrupções, para garantir a boa leitura das horas, estas são lidas duas vezes. Caso sejam diferentes lê-se do RTC novamente as horas sabendo que estas agora estão corretas. (Menu.c l.217 a l.246)

Como o relógio estava a enviar os dados das horas no formato BCD, foi importante informar o RTC que desejávamos os dados em binário, sendo assim facilitada a utilização das horas. (RTC.c l.22 a l.30)

Descrição dos módulos

BMP.h - Contém a declaração das funções relativas ao carregamento de imagens bitmap assim como a definição de structs uteis para o manuseamento das imagens.

BMP.c - Contém a implementação das funções declaradas em BMP.h.

Graficos.h - Contém a declaração das funções relativas à manipulação de imagens, como animações e preparação dos frames e variáveis uteis para a implementação das funções.

Graficos.c - Implementação das funções declaradas em Graficos.h

Jogo.h - Contém a declaração das funções relativas ao jogo e as ações no jogo assim como duas variáveis uteis para a implementação das funções.

Jogo.c - Implementação das funções declaradas em Jogo.h.

Menu.h - Contém a declaração de todas as funções relativas ao menu inicial, ao menu de máxima pontuação, ao relógio e uma struct com as respetivas imagens de fundo dos menus, bem como o estado do jogo.

Menu.c - Contém a implementação de todas as funções declaradas em Menu.h.

RTC.h - Contém a declaração de todas as funções relativas ao RTC.

RTC.c - Implementação das funções declaradas em RTC.h.

Utilidades.h - Contém a declaração de várias funções uteis para a iniciação e término do jogo e o carregamento e gravação da pontuação máxima.

Utilidades.c - Implementação das funções declaradas em Utilidades.h.

kbd.h - Contém a declaração de todas as funções relativas ao teclado.

kbd.c - Implementação de todas as funções relativas ao teclado e declaradas em kbd.h.

macros.h - Contém as definições de todas as macros utilizadas no jogo.

main.c - Contém a implementação do código da função main.

menufinal.h - Contém a declaração de várias funções relativas ao menu de final de jogo, pontuação e uma struct com a imagem de fundo do menu e a pontuação do jogador.

menufinal.c - Implementação das funções declaradas no menufinal.h.

mouse.h - Contém a declaração de todas as funções relativas ao rato e uma struct que contém duas variáveis indicando as coordenadas do rato, uma imagem para o rato e um inteiro que verifica se o botão esquerdo do rato está pressionado.

mouse.c - Implementação de todas as funções declaradas em mouse.h

timer.h - Contém a declaração de todas as funções relativas ao timer.

timer.c - Implementação de todas as funções declaradas em timer.h.

vbe.h - Variáveis uteis para a inicialização e fecho do modo de vídeo.

Divisão de trabalho

Nós dividimos igualmente o projeto, onde o Jorge Ferreira ficou encarregue dos Menus e da leitura de ficheiros bem como das pontuações e o Nuno Silva ficou encarregue do jogo e mais da parte gráfica, das animações do ninja e dos fundos.

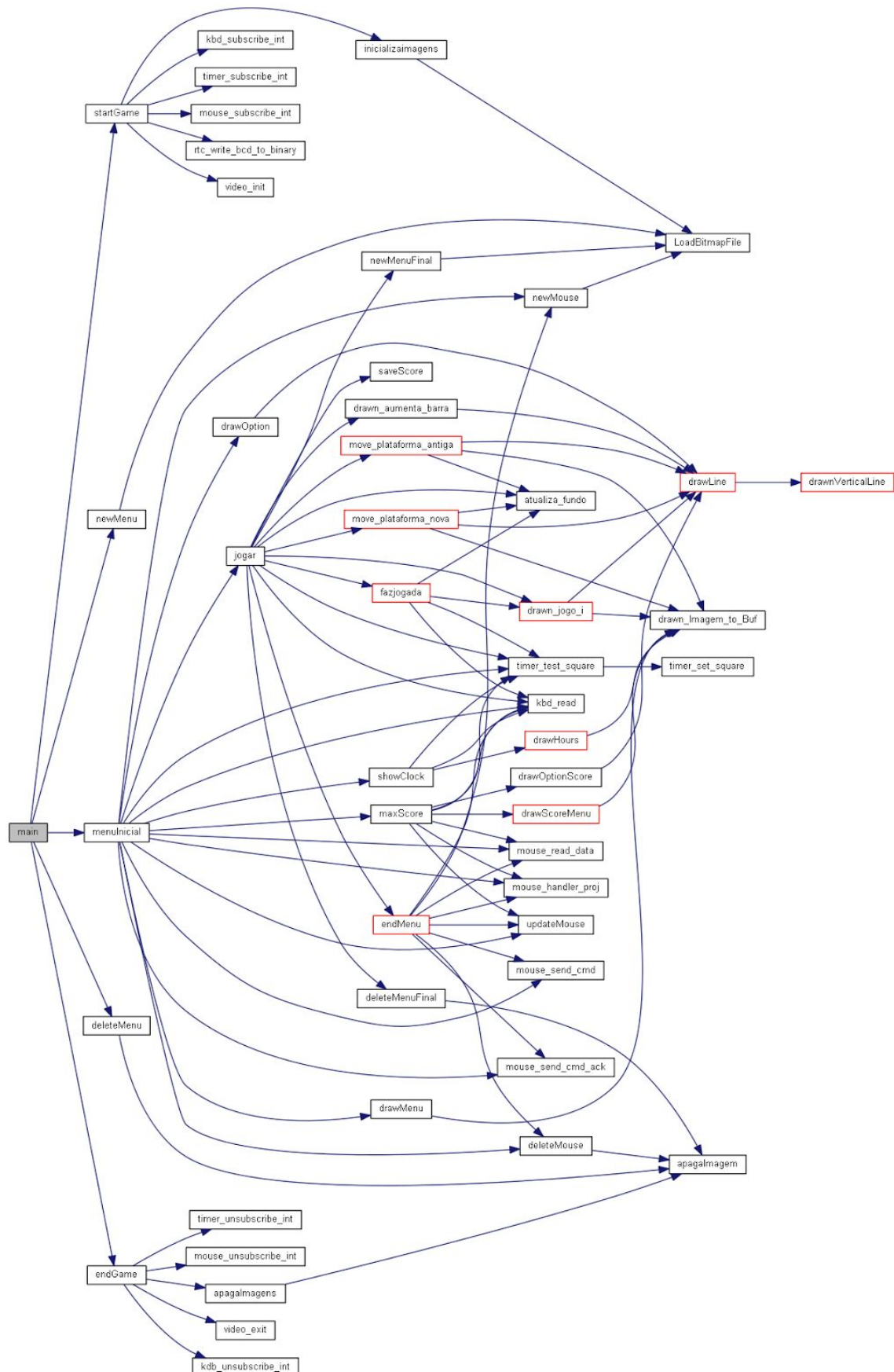
A divisão serviu para um desenvolvimento fluido, mas havia sempre bastante interação entre módulos onde cada um ajudava o outro sempre que precisava ou tivesse uma ideia nova.

Em suma, existiu sempre bastante interação e houve trabalho de ambos em todo o projeto.

Tanto o relatório como o projeto foram realizados em conjunto usando o teamspeak para comunicação por voz e na realização do relatório, também se utilizou o google docs para o manuseamento do ficheiro do relatório, onde era possível editar o mesmo ficheiro em tempo real.

Diagrama de chamadas

Diagrama principal:



Detalhes da Implementação

Em várias partes do jogo foram usados materiais lecionados nas aulas bem como outros externos.

Um dos aspectos mais importantes em toda a aplicação é a leitura de imagens. Para isso utilizamos um código não realizado por nós que faz o carregamento de imagens no formato BMP.

Outro aspecto importante usado foi a geração de frames pelo timer. Com a geração de uma frequência correta, é possível obter “frames per second” constantes e esta geração foi útil em toda a aplicação, tanto nos menus como no jogo em si, pois é a base de toda a aplicação e permitiu-nos ter um maior controlo nas animações, sendo que usamos macros para regular várias animações, como por exemplo velocidade do movimento das nuvens, ou a velocidade do movimento do andar do ninja.

As structs foram também utilizadas em certas partes da aplicação. Por exemplo, no rato é útil a utilização de uma struct que contém as coordenadas do rato sempre atualizadas, uma imagem do respetivo rato carregada quando se faz alocação da memória para a struct e uma outra variável para verificar se o botão esquerdo do rato está a ser pressionado. Esta última variável é essencial para verificar, nos menus, se o utilizador da aplicação pretende entrar numa das opções apresentadas.

A leitura de ficheiros bem como a sua escrita também foi usada na aplicação. Para guardarmos sempre a pontuação máxima de todo o jogo, bem como atualizá-la foi utilizado, como base de dados, um ficheiro externo de texto de nome “score”. Sempre que o jogo acaba e se a pontuação da respetiva jogada for superior a pontuação máxima existente no ficheiro, esta é atualizada. Caso a pontuação não seja superior, a pontuação máxima existente no ficheiro é mantida. Para mostrar no menu da pontuação é apenas usado o carregamento da pontuação máxima já existente no ficheiro.

Um outro periférico utilizado foi o RTC.

No início estávamos a usar o RTC com interrupções e funcionava parcialmente mas quando saíamos do menu inicial e voltávamos ao menu, deixava de funcionar e então decidimos não usar com interrupção no final visto que o resto do projeto já estava concluído.

Não usando interrupções, existiriam sempre problemas em saber se as horas recebidas estariam corretas pois ao ler entre, por exemplo, os segundos e as horas existiria um delay pequeno, mas que era o bastante para que nesse pequeno tempo os segundos se alterassem e afetassem as horas. Para contornar esse aspecto, são lidas as horas numa primeira situação, depois utilizando um buffer são lidas novamente as horas. Caso as horas sejam

diferentes é porque existiu atualização da hora nesse intervalo, e não sabendo qual delas é a correta, é lida novamente a hora, sabendo que esta última é a correta.

Como o RTC nos dá os números em BCD, era essencial para melhor interpretação, bem como para facilitar a amostragem das horas no menu do relógio a sua passagem para binário. Para isso, pediu-se ao RTC para toda a informação fornecida por ele ser em binário.

O jogo tem várias animações que tornam o jogo mais agradável, as nuvens estão sempre a mover-se, as ervas só se movem quando as plataformas se movem dando uma maior sensação de movimento, o próprio ninja tem a animação de andar e de cair quando perde o jogo, as plataformas também se movem. Existe uma animação onde a plataforma que o ninja conseguiu alcançar se move para o lado esquerdo, a posição inicial, a barra também está presente para que o ninja fique sempre pousado em algo firme e não fique a voar em plataformas pequenas. Existe também uma animação para a geração da nova plataforma que é gerada aleatoriamente, onde ela aparece e vai se deslocando até à posição que também é gerada aleatoriamente. Nós editamos e criamos as imagens para tornar as animações possíveis.

Conclusão

Um dos aspetos negativos da disciplina é a sua dificuldade e o limitado número de professores/monitores para as aulas práticas.

Um dos aspectos positivos é a possibilidade de poder escolher o tipo de projeto, sendo que um jogo foi a nossa escolha pois nunca tínhamos feito nada do gênero e tornou-se uma aventura na qual nunca tínhamos participado e sempre quisemos participar.

Apesar da dificuldade da disciplina foi importante utilizar e interligar os diferentes periféricos e contornar as suas limitações. Recolhemos diversos conhecimentos ao nível de hardware e da sua ligação com a programação.

Auto-Avaliação

Existiu uma contribuição ativa por parte de cada indivíduo do grupo, o que nos levou a concluir que a % de contribuição para o projeto por parte de Jorge Ferreira e de Nuno Silva foi respetivamente 50, 50.

Bibliografia

<http://stackoverflow.com/questions/14279242/read-bitmap-file-into-structure>

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

<http://web.fe.up.pt/~pfs/aulas/lcom2015/>