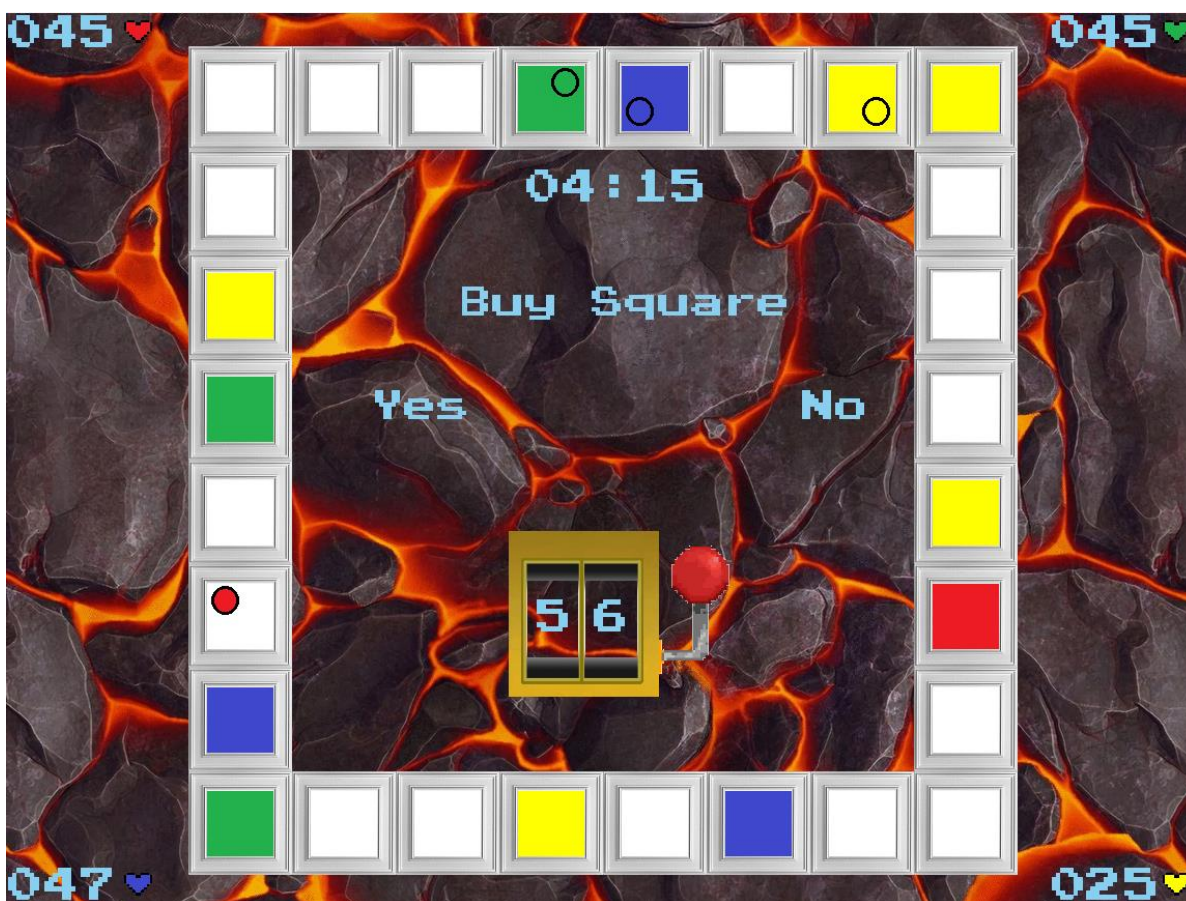


MINIX IS LAVA



Laboratório de Computadores 2019/20

Turma 7 – Grupo 1

Janeiro 2020

Carlos Daniel Rodrigues Lousada

up201806302@fe.up.pt

Miguel Augusto Marques da Silva

up201806388@fe.up.pt

Índice

Instruções de Utilização	2
Menu Inicial	2
Help	3
Start	4
Jogo	4
Game Over	6
Estado do Projeto	7
Dispositivos Usados.....	7
Timer.....	7
Teclado.....	7
Rato	7
Placa Gráfica.....	8
RTC(Real Time Clock).....	8
Organização e Estrutura de Código	9
Proj.....	9
i8254.....	9
i8042.....	9
Timer.....	9
Teclado.....	9
Rato	9
Graphics.....	9
Utils	10
RTC (Real Time Clock).....	10
Drawings	10
Call Graphs.....	11
Detalhes de Implementação	12
Conclusões	13

Instruções de Utilização

Menu Inicial

Ao inicializar o programa, é mostrado ao utilizador o main menu, com as opções:

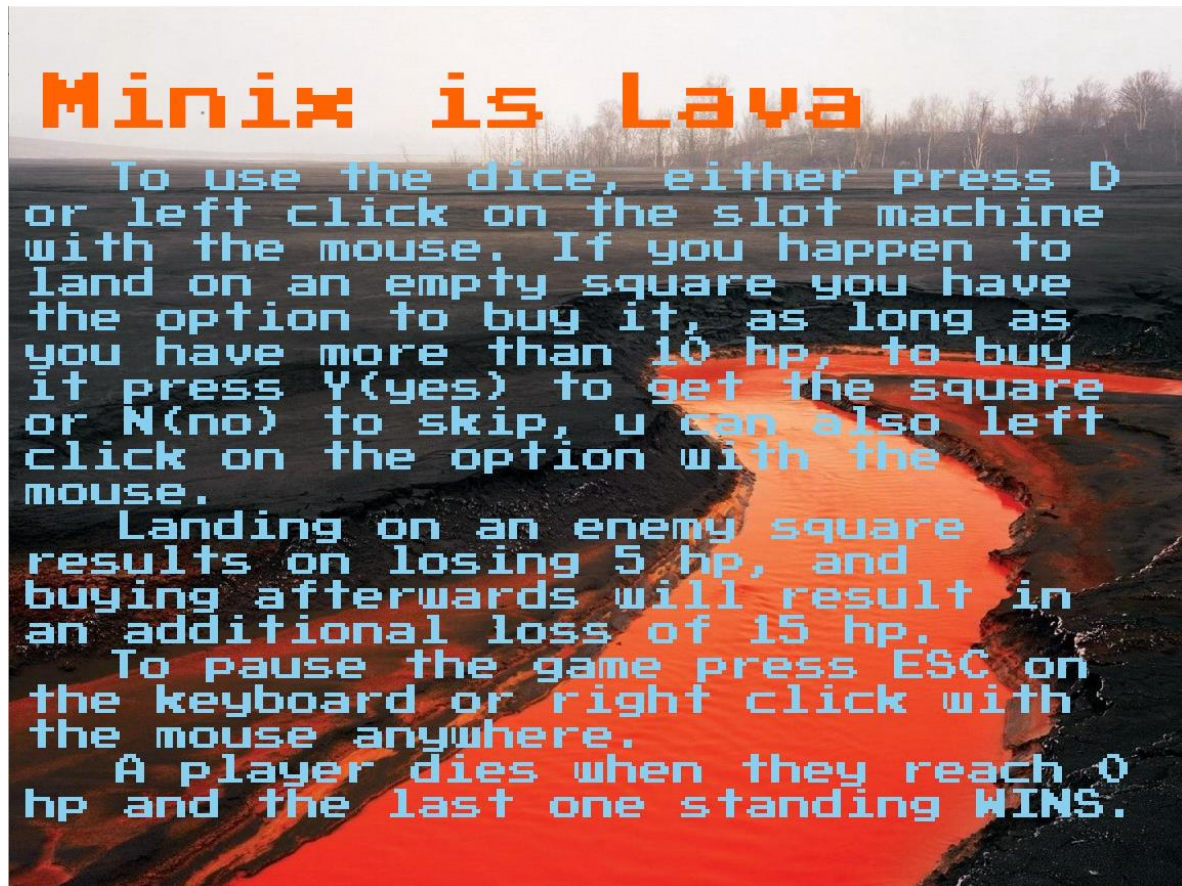
- Start, para iniciar um novo jogo;
- Help, redirecionando o utilizador para um menu de ajuda, com informações sobre como jogar;
- Exit, para fechar o programa.



Para navegar no menu, o utilizador tanto pode utilizar as setas (para cima e baixo), ou o rato, confirmando a opção escolhida clicando no ENTER ou com o botão esquerdo do rato.

Help

Este menu apresenta instruções sobre como jogar. Algumas opções do mouse não se refletem no jogo, pois não conseguimos dar debug a algumas funcionalidades do mouse perto da data de entrega.



Start

Jogo

Passando por este menu, é então iniciado o jogo. Este é uma mistura do Monopólio e do Floor Is Lava. Cada um dos 4 jogadores começa com 100 hp e joga-se por turnos. O utilizador é, por default, a bola vermelha. Assim, o jogador lança um dado, clicando na tecla D, aparecendo numa slot machine os números referentes ao dado, movendo-se o jogador automaticamente para a casa correspondente. Ao chegar a essa casa é aberto um “menu de compra” onde o jogador pode escolher ou não comprar a casa, pintando-a. Se não a comprar, anda acontece, caso compre, temos vários cenários. Ou a casa já estava ocupada, pelo que o jogador perde 5 hp por a pisar e depois perde 15 hp se decidir comprá-la. Se a casa não estava ocupada, o custo da compra é de 10 hp.



O último sobrevivente no tempo definido vence. Caso o jogador perca é redirecionado para um menu de Game Over, sendo novamente redirecionado para o menu inicial.

Se o jogador, durante o jogo, clicar na tecla ESC, este é direcionado para um menu de pausa onde pode voltar ao jogo, clicando no ENTER, ou voltar ao menu clicando no ESC.



Game Over

Quando o jogador perde, ou desiste (voltando ao menu quando em pause), é redirecionado para a screen de game over durante 3 segundos, onde pode ver o hp de todos os jogadores ao acabar o jogo, antes de voltar ao main menu.



Estado do Projeto

Dispositivos Usados

Dispositivo	Funcionalidade	Interrupções
Timer	Atualização do estado do jogo (criação da slot machine, players, etc)	Sim
Teclado	Navegação nos menus e inputs de jogo (D para lançar dados, ESC para pausar, etc)	Sim
Rato	Navegação no jogo	Sim
Placa Gráfica	Desenho e visualização dos menus e do jogo	Não
RTC	Obtenção da data e horas atuais	Não

Timer

Este dispositivo tem como função permitir o uso do double buffering, implementado com o memcpy, que, por sua vez, suaviza a criação e o movimento de sprites no ecrã. Assim, o timer é responsável por atualizar as posições dos vários sprites existentes bem como para contabilizar o tempo de jogo escolhido pelo utilizador.

Assim, o timer é um dos módulos mais importantes do jogo dado ser este que atualiza constantemente o estado do jogo, chamando as funções necessárias no tempo correto.

Teclado

Este dispositivo é utilizado para a navegação dos menus, bem como a maior parte das ações do utilizador no jogo. Permite a saída do jogo por parte do utilizador, clicando no ESC, bem como atirar os dados, pausar, etc. É assim um dos dispositivos mais importantes relativamente à jogabilidade. As suas funções auxiliares encontram-se no **keyboard.c**.

Rato

O rato é utilizado durante o jogo, permitindo ao jogador selecionar se pretende ou não comprar a casa onde calhou, como alternativa ao teclado, sendo também importante para uma jogabilidade diversificada. As suas funções foram importadas do lab4, encontrando-se no **mouse.c**.

Placa Gráfica

Este dispositivo é usado para o processamento gráfico do jogo, mostrando todas as imagens vistas no ecrã. O modo de vídeo escolhido para o projeto foi o 0x14C, com a resolução de 1152x864 pixéis, usando direct color, tendo 32 bits por pixel.

De forma a tornar as transições mais fluidas e agradáveis, foi implementada a técnica de double buffering. Esta técnica tornou a animação de sprites, como por exemplo da do rato muito mais suaves.

A criação das sprites e de todas as imagens vistas no ecrã são feitas no ficheiro **drawings.c**, o qual contém funções como `draw_background()`, `create_sprite()`, `animate_sprite()`, entre outras.

RTC(Real Time Clock)

Este dispositivo é usado para a obtenção da data e horas atuais, com funções do (**rtc.c**). A data e hora obtida é então apresentada e atualizada durante o jogo.

Organização e Estrutura de Código

Proj

O ficheiro proj.c é a base do projeto, sendo aí onde se encontram os loops importantes que permitem o desenrolar do programa. É aqui que são tratadas todas as interrupções e opções do utilizador.

i8254

Módulo importado das aulas práticas relativas aos labs 2, 3, 4 e 5. Contém as constantes usadas pelo timer.

i8042

Módulo importado das aulas práticas relativas aos labs 3, 4 e 5. Contém constantes usadas pelo keyboard, mouse e rtc.

Timer

Módulo importado do lab2. Contém as funções que permitem tratar as interrupções do timer bem como dar subscribe, unsubscribe e funções relacionadas com o data reporting.

Teclado

Módulo importado do lab3. Contém funções para dar subscribe e unsubscribe dos interrupts, bem como para o tratamento de interrupções.

Rato

Módulo importado do lab4. Contém funções que permitem tratar as interrupções bem como dar subscribe, unsubscribe e data reporting. Tem também funções que permite dar parse nos packets provenientes das interrupções.

Graphics

Módulo importado do lab5. Contém funções que permitem entrar e sair do modo gráfico, bem como o mapeamento de endereços de memória de vídeo.

Utils

Contém a função `util_sys_inb()`, que permite usar variáveis de 8 bits num `sys_inb` call, não precisando de dar cast para `uint32_t`.

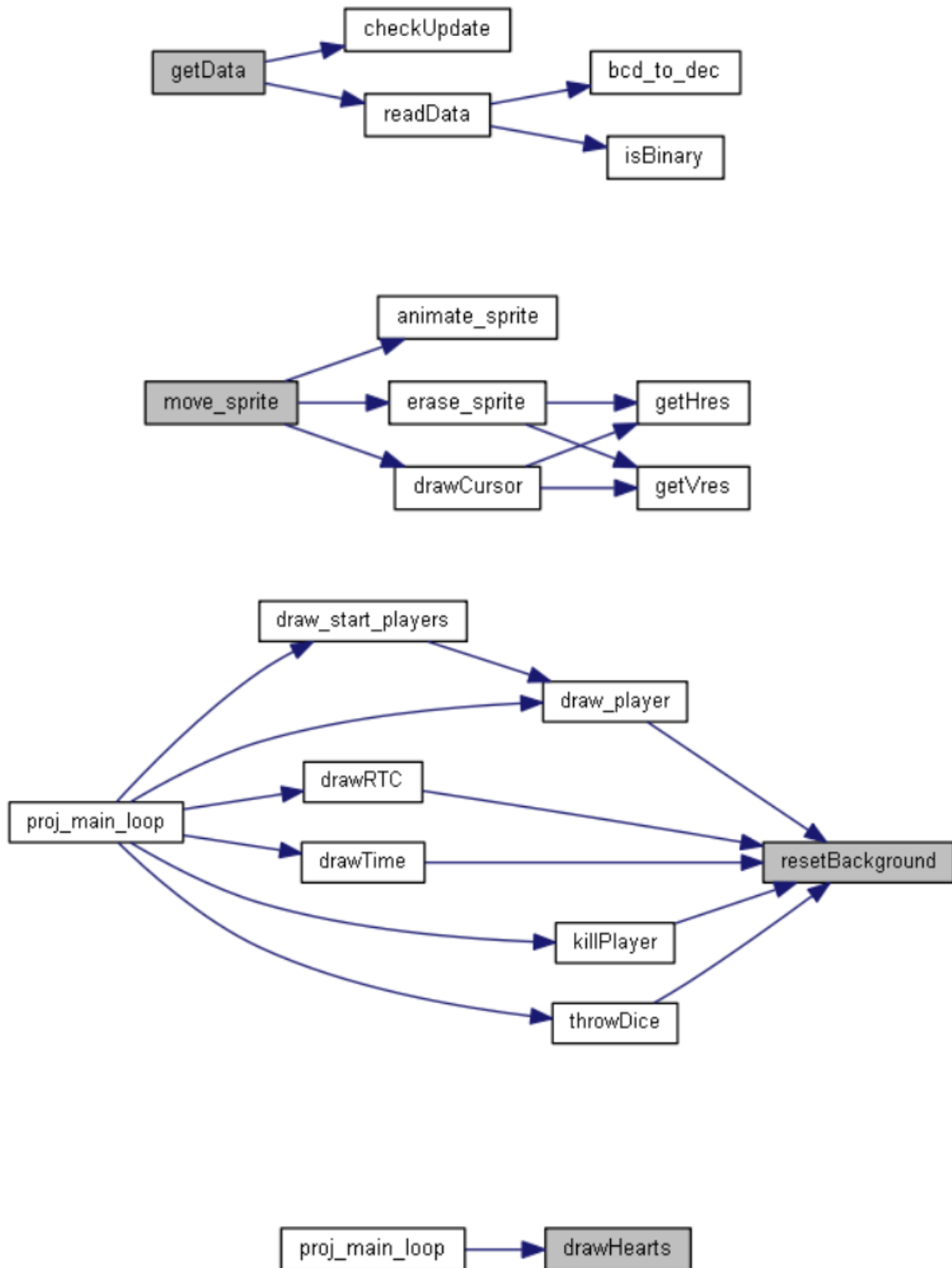
RTC (Real Time Clock)

Módulo contém as funções necessárias à implementação do RTC. Permite a obtenção da data e hora atual através do `getData()`, a informação é guardada na struct `time_date`.

Drawings

Módulo que contém todas as funções relacionadas com o desenho de sprites/objetos. Usando a função `move_sprite()`, que, por sua vez chama as funções `erase_sprite()`, `animate_sprite()` e `draw_sprite()`, as quais permitem o movimento das sprites pelo ecrã.

Call Graphs



Detalhes de Implementação

Consideramos que a implementação da maioria dos módulos foi um sucesso, para além do mouse e da serial port. Esta última não foi sequer implementada devido à dificuldade encontrada na tentativa de atualizar os ecrãs de ambos os PC's ao mesmo tempo.

Em relação ao rato, acreditamos que devido ao nosso descuido inicial aquando do desenvolvimento do código em camadas - o que é evidente nos call graphs acima - não permitiu um debug eficaz deste dispositivo, pelo que nos levou bastante tempo a melhorar a suavidade do mouse no ecrã, não nos permitindo implementar todas as funções que seriam de esperar dele.

Uma das maiores dificuldades encontradas, excluindo as acima referidas, foi a constante existência e movimentação de sprites durante o jogo, pelo que demoramos bastante nas funções de `erase_sprite()` e `resetBackground()` nomeadamente.

Conclusões

Para finalizar, achamos que a unidade curricular de LCOM foi a que se provou mais difícil ao longo do semestre, sendo aquela na qual depositámos mais horas semanais.

Inicialmente as dificuldades eram enormes, mas, ao longo do tempo e com mais prática acreditamos que a abordagem aos temas dos labs, do projeto e o conhecimento em geral dos periféricos se tornaram mais simples.

Relativamente ao projeto em si, achamos que foi um bom desafio, que poderia ter corrido melhor caso tivéssemos gerido melhor o tempo, deixando muitas “pequenas” coisas para o fim, o qual acabou por ser fatal, não nos permitindo organizar o código da forma que pretendíamos, tornando o seu debug e leitura muito mais difíceis.

Em suma, apesar da grande dificuldade da cadeira e do projeto, acreditamos que esta teve um impacto bastante positivo na nossa formação, permitindo-nos desenvolver competências em relação aos vários periféricos e numa linguagem de baixo nível como C, permitindo-nos também perceber que a organização inicial do código é fundamental para um desenvolvimento saudável do projeto, o que poderia ter levado a um total funcionamento do rato.