|  |
| --- |
| **Projecto Final de LCOM -Minesweeper** |
|  |
|  |
|  |
|  |
| **Mariana Oliveira – up201207835**  **Pedro Castro – up201305337** |
| **Ano Lectivo 2014/2015** |
|  |



Índice

[Introdução 3](#_Toc407987424)

[Descrição 3](#_Toc407987425)

[Instruções de utilização do jogo 4](#_Toc407987426)

[Menu Principal 4](#_Toc407987427)

[Menu Multi Player 4](#_Toc407987428)

[Modo de Jogo Single Player 5](#_Toc407987429)

[Modo de Jogo Multi Player 5](#_Toc407987430)

[Multi Player Local 5](#_Toc407987431)

[Multi Player Entre PCs 6](#_Toc407987432)

[Periféricos 6](#_Toc407987433)

[Módulos 7](#_Toc407987434)

[Teclado 7](#_Toc407987435)

[Rato 7](#_Toc407987436)

[Modo Gráfico 8](#_Toc407987437)

[RTC 8](#_Toc407987438)

[Timer 8](#_Toc407987439)

[Menu 9](#_Toc407987440)

[Bitmap 9](#_Toc407987441)

[Porta de Série 9](#_Toc407987442)

[Jogo 10](#_Toc407987443)

[Conhecimentos requeridos 11](#_Toc407987444)

[Avaliação 11](#_Toc407987445)

[Disciplina 11](#_Toc407987446)

[Grupo 11](#_Toc407987447)

[Conclusão 11](#_Toc407987448)

# Introdução

De acordo com o funcionamento da disciplina de Laboratórios de Computadores (LCOM), a nossa proposta para o projecto final é o jogo Minesweeper.

Assim, o nosso projecto está estruturado em módulos de modo a ser mais fácil e eficaz a análise do mesmo.

# Descrição

Este jogo, jogado com um ou dois jogadores, tem como objectivo encontrar todas as casas que não sejam as bombas escondidas no tabuleiro.

No tabuleiro existem três tipos de “casas”: as bombas, aleatoriamente espalhadas pelo tabuleiro; as “casas” em branco e as “casas” com números, números estes que correspondem ao número de bombas existentes na vizinhança dessa casa.

Podem ainda existir “casas” com bandeiras, caso o jogador o pretenda, para assinalar as “casas” em que pensa existir bombas.

Em modo *single player*, o jogador tem um limite de tempo, isto é, se não encontrar todas as casas que não são bombas antes do final do tempo, perde.

Em modo *multi player*, em que pode ser jogo local ou em diferentes computadores, começa o jogador vermelho e, após a sua jogada, joga o jogador azul e assim sucessivamente.

Caso um dos jogadores demore mais do que 30 segundos numa jogada, perde o poder de jogo. Por outro lado, se acertar numa “casa” com bomba, o timer recomeça e pode jogar de novo.

O jogo acaba quando um dos jogadores descobre mais de metade das bombas existentes e é esse jogador o vencedor.

# Instruções de utilização do jogo

## Menu Principal



## 

No menu principal há duas opções de jogo: SINGLEPLAYER e MULTIPLAYER.

Assim como a opção de sair do jogo, clicando na cruz “X” no canto superior esquerdo ou na tecla ESC.

## multi player.pngMenu Multi Player

No menu Multi Player existem três opções: LOCAL, opção esta que permite jogarem dois jogadores, à vez, no mesmo computador; HOST, opção que permite jogarem dois jogadores em computadores diferentes, sendo este o host do jogo; e por fim, a opção JOIN que permite ao jogador juntar-se ao jogo criado por outro jogador, noutro computador.

## Modo de Jogo Single Player lost single player.pngSingle Player.png

No modo Single Player, o jogador perde se clicar numa casa com bomba e, aí, é-lhe revelada a matriz com a localização de todas as bombas existentes na matriz.

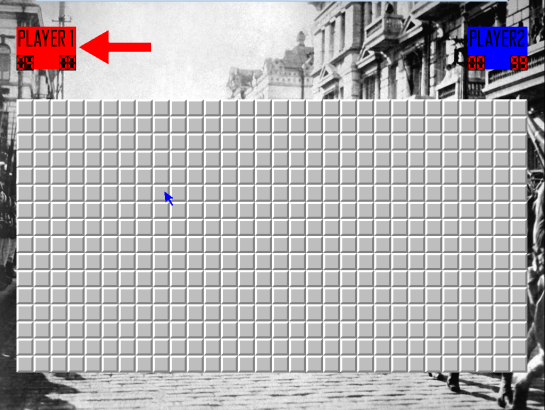
## Modo de Jogo Multi Player

### Multi Player Local

### multi player local.png

No modo Multi Player Local, os jogadores jogam à vez, em que cada um tem um cronómetro que decresce a partir dos 30 segundos, caso acerte numa casa com um número, passa para outro jogador e o cronómetro reinicia; caso acerte numa casa com bomba, o cronómetro reinicia e pode jogar novamente. Caso os 30 segundos passem, esse jogador perde o poder de jogo.

### Multi Player Entre PCs



No modo Multi Player entre PCs, enquanto se espera que ambos os jogadores estejam conectados, aparece a primeira imagem com a mensagem “CONNECTING”.

Após ambos os jogadores estarem ligados, o jogo inicia-se com o jogador HOST (jogador vermelho), tendo uma seta com a cor respectiva a apontar para o seu cronómetro e, após este, joga o jogador azul e assim sucessivamente.

Ganha o jogador que descobrir mais de metade das bombas existentes na matriz.

Caso um dos jogadores saia do jogo, aparece a terceira imagem, com a mensagem “DISCONNECTED”, terminando o jogo, em ambos terminais.

# Periféricos

|  |  |  |
| --- | --- | --- |
| Nome | Função | Interrupções/Polling |
| Porta de Série | Jogo entre PCs | P |
| Rato | Seleccionar opções nos menus e “casas” no jogo | I |
| RTC | Display da hora | P |
| Teclado | Sair de menus/jogo | I |
| Timer | Controlar o tempo de jogo/jogada | I |

# Módulos

## Teclado

O módulo do teclado (ModTeclado) é utilizado para se poder sair do jogo e dos menus, premindo a tecla ESC.

Funções utilizadas neste módulo:

* **void kbd\_int\_handler**();
* **int kbd\_unsubscribe\_int**();
* **int kbd\_subscribe\_int**(**void**);

O responsável por este módulo era o aluno Pedro Castro, e a contribuição e a participação do mesmo foi de 50%, sendo o restante atribuído à aluna Mariana Oliveira. A importância deste módulo para o projecto é de cerca de 5%.

## Rato

O módulo do rato (ModRato) é basicamente usado em todo o jogo.

Este permite percorrer as opções do menu e selecciona-las, efectuar as jogadas: botão esquerdo - pressiona a “casa” e revela o seu conteúdo, botão direito – coloca uma bandeira na “casa” seleccionada. Permite também a saída dos menus, pressionando o “X” situado no canto superior esquerdo.

Funções utilizadas neste módulo:

* Rato\* **getRato**();
* Rato\* **newRato**();
* **int mouse\_subscribe\_int**();
* **int mouse\_unsubscribe\_int**();
* **int mouse\_int\_handler**();
* **void updateMouse**();
* **int first\_byte**(**unsignedlong** mouse\_byte);
* **void drawRato**(**int** cor);
* **int enable\_packets**();

O responsável por este módulo era a aluna Mariana Oliveira, e a contribuição e a participação da mesma foi de 60%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 20%.

## Modo Gráfico

O módulo do modo Gráfico (ModGrafico) permite a a troca entre buffers e a sua alteração.

Funções utilizadas neste módulo:

* **void trocarRato\_buffer**();
* **void trocarVideo\_Mem\_Rato**();
* **int vg\_exit**(**void**);
* **void** \***vg\_init**(**unsignedshort** mode);
* **void update\_screen**(**int** cor,**int** turn,**int** relog);

O responsável por este módulo era a aluna Mariana Oliveira, e a contribuição e a participação da mesma foi de 40%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 20%.

## RTC

O módulo do RTC (ModRTC) tal como o próprio nome indica, permite a visualização da hora actual, no menu principal do jogo e no menu Multi Player.

Funções utilizadas neste módulo:

* **int scan\_rtc** (**unsignedlong** index, **unsignedlong** \*dado);
* **int write\_rtc**(**unsignedlong** index, **unsignedlong** dado);
* **void disable**();
* **void enable**();
* **void wait\_valid\_rtc**(**void**);
* **unsigned char bcd\_to\_binary**(**unsignedchar** bcdvalue);
* **unsigned char binary\_to\_bcd**(**unsignedchar** binaryvalue);
* **int rtc\_date**();

O responsável por este módulo era o aluno Pedro Castro, e a contribuição e a participação da mesma foi de 50%, sendo o restante atribuído à aluna Mariana Oliveira. A importância deste módulo para o projecto é de cerca de 5%.

## Timer

O módulo do Timer (ModTimer) é utilizado para cronometrar jogadas, tanto no modo single player como no modo multi player.

Funções utilizadas neste módulo :

* **int timer\_subscribe\_int**();
* **int timer\_unsubscribe\_int**();
* **void timer\_int\_handler**();

O responsável por este módulo era a aluna Mariana Oliveira, e a contribuição e a participação da mesma foi de 50%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 10%.

## Menu

O módulo do Menu (Menu) tal como o próprio nome indica é o módulo onde são construídos os menus de jogo, o primeiro com as opções “Single Player” e “Multi Player” , e o segundo, quando a opção “Multi Player” é seleccionada, com as opções “Local Game”, “Host” and “Join”.

Funções utilizadas neste módulo :

* **void draw\_main\_menu**();
* **void multi\_menu**(**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse);
* **void draw\_multi\_menu**();
* **void delete\_relogio**();
* **void load\_relogio**();

O responsável por este módulo era a aluna Mariana Oliveira, e a contribuição e a participação da mesma foi de 40%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 10%.

## Bitmap

O módulo Bitmap (modBitmap) é utilizado para se carregar e desenhar os bitmaps correspondentes ao nosso jogo.

Funções utilizadas neste módulo :

* Bitmap\* loadBitmap(const char\* filename);
* Void drawBitmap(Bitmap\* bitmap, int x, int y, Alignment alignment, uint16\_t\* temp\_buffer);
* void deleteBitmap(Bitmap\* bmp);

O código correspondente a este módulo foi retirado, na sua integridade, do blog do aluno Henrique Ferrolho.

## Porta de Série

O módulo da Porta de Série (modPortaSerie) é o módulo responsável por tornar o jogo Multi Player, através da ligação entre dois PCs através da Porta de Série.

Funções utilizadas neste módulo :

* **void changeDLAB**(**unsignedshort** base\_addr,**int** bit);
* **int getChar**(**unsignedshort** base\_addr, **unsignedchar** \*c);
* **int getCharOne**(**unsignedshort** base\_addr, **unsignedchar** \*c);
* **int sendChar**(**unsignedshort** base\_addr,**char** c);
* **void setPorta**(**unsignedshort** base\_addr, **unsignedlong** bits, **unsignedlong** stop, **long** parity, **unsignedlong** rate);

Os responsáveis por este módulo eram ambos os membros do grupo,sendo que a contribuição e a participação da aluna Mariana Oliveira foi de 30%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 10%.

## Jogo

O módulo Jogo (modJogo) é o módulo onde todos os outros módulos são requeridos, isto é, este módulo cria o jogo em si, com a respectiva matriz.

A localização e o número de bombas são calculados aleatoriamente e as restantes casas são criadas de acordo.

Funções utilizadas neste módulo :

* **void print\_table**(Mine\*\* table);
* Mine\*\* **create\_table**(**int** difficulty);
* **void create\_interrupts**(**int** \*irq\_set\_timer,**int** \*irq\_set\_keyboard,**int** \*irq\_set\_mouse);
* **void stop\_interrupts**();
* **int post\_game\_state\_multi**(**int** difficulty,**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse,**int** jogador);
* **int jogo\_multi\_player**(**int** difficulty,**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse);
* **int right\_click\_screen**(Mine\*\*\* table, **int** x, **int** y, **int** difficulty);
* **void click\_vazio**(Mine\*\*\* table, **int** k, **int** j, **int** difficulty,**int** \*por\_carregar,**int** single);
* **int click\_screen**(Mine\*\*\* table, **int** x, **int** y, **int** difficulty, **int** \*filled,**int** \*por\_carregar,**int** single);
* Mine\*\* **fill\_table**(Mine\*\* table,**int** difficulty,**int** k, **int** j,**int** single,**char** \*seed,**int** use); /// use é parausar a seed paracriar
* **int post\_game\_state**(**int** difficulty,**int** win,**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse,Mine\*\* table);
* **int jogo\_single\_player**(**int** difficulty,**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse);
* **void update\_cronometro**(**int** tempo);
* **void load\_cronometro**();
* **void delete\_cronometro**();
* **void update\_multi\_cronometro**(**int** tempo, **int** jogador,**int** jogador1,**int** jogador2);
* **void draw\_end\_scree\_multi\_porta**(**int** turn,**int** jogador);
* **void end\_scree\_multi\_porta**(**int** irq\_set\_timer,**int** irq\_set\_keyboard,**int** irq\_set\_mouse,**int** turn,**int** jogador);

Os responsáveis por este módulo eram ambos os membros do grupo, sendo que a contribuição e a participação da aluna Mariana Oliveira foi de 30%, sendo o restante atribuído ao aluno Pedro Castro. A importância deste módulo para o projecto é de cerca de 10%.

# Conhecimentos requeridos

* Máquinas de Estado

As máquinas de estado, apesar de ter sido falado nas aulas, requereram uma investigação mais profunda pela nossa parte, de modo a poderem ser utilizadas no nosso projecto, por exemplo, nos vários menus.

* Triple Buffering

O triple buffering, apesar de não ter sido mencionado nas aulas, foi uma parte essencial do nosso projecto, no sentido em que nos permite manter parte da frame inalterada, por exemplo, apesar de o facto estar a ser constantemente desenhado, nada mais alem disso se altera na frame.

# Avaliação

## Disciplina

A avaliação da disciplina será feita a partir de três parâmetros: interesse suscitado aos alunos, utilidade e funcionamento da disciplina.

No primeiro parâmetro a nossa opinião é positiva, no sentido em que, tanto nas aulas teóricas, como práticas, estas foram leccionadas de forma ao aluno não perder interesse nas matérias dadas assim como interessar-se por saber mais.

No segundo parâmetro a nossa opinião também é positiva visto que é das disciplinas mais relacionadas com programação e, obviamente, muito útil para o nosso curso.

Por fim, no terceiro parâmetro a nossa opinião não é tão positiva, mais no sentido das aulas práticas e o respectivo tempo para os laboratórios, isto é, algumas turmas tinham menos tempo que outras para a realização das actividades de laboratório, tempo este relativamente significativo visto que chegava até aos 3 dias de diferença.

## Grupo

A avaliação de grupo é feita a partir de dois parâmetros: contribuição e participação.

Assim, a avaliação da aluna Mariana Oliveira na contribuição e na participação é de 40%, sendo o restante atribuído ao aluno Pedro Castro.

# Conclusão

No final do projecto é possível afirmar que este foi bem sucedido e que as metas propostas foram cumpridas.