



# PONG

Projeto final LCOM

**U.**PORTO

**FEUP** FACULDADE DE ENGENHARIA  
UNIVERSIDADE DO PORTO

Frederico Lopes e Luís Lucas

FEUP

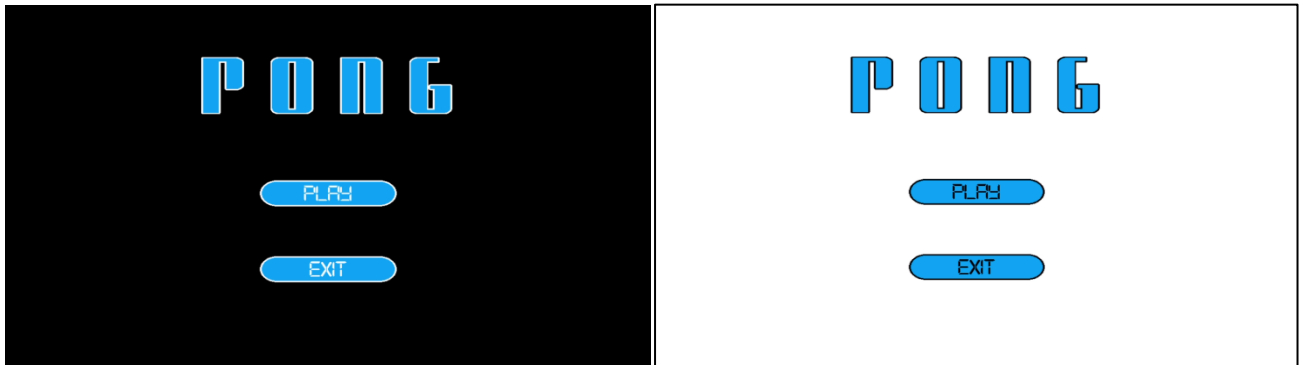
# Índice

Instruções de utilização .....	2
Main Menu.....	2
Pong .....	2
Winner .....	3
Estado do projeto .....	4
Timer .....	4
Keyboard .....	4
Mouse .....	4
Graphics card .....	4
RTC .....	5
Organização e estrutura do código .....	6
Game .....	6
Utils .....	<b>Error! Bookmark not defined.</b>
Timer .....	6
Keyboard .....	6
Mouse .....	6
Video .....	6
Proj.....	6
Function call graph.....	7
Detalhes de implementação.....	8
Conclusões.....	9

# Instruções de utilização

## Main Menu

Quando o jogo é iniciado é apresentado um menu principal, este menu é controlado com o rato e tem dois “botões” que podem ser selecionados, o EXIT para sair do jogo e o PLAY para iniciar o jogo. Tal como vai acontecer no resto do projeto, existem dois menus diferentes, um que é apresentado quando é dia e outro que é apresentado de noite. A diferença entre estes é a cor do rato do fundo, dos botões e o contorno das letras com o nome do jogo como se pode verificar nas imagens.

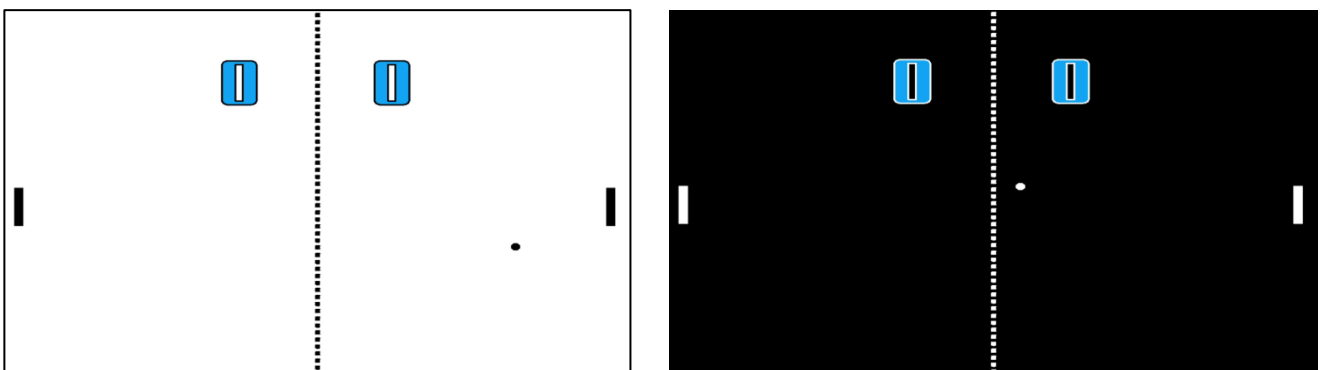


## Pong

Depois de ter selecionado o botão play o jogo começa, este jogo é multijogador e o objetivo é fazer com que a bola chegue ao outro lado do ecrã, marcando assim ponto na baliza adversaria.

Para mover as raquetes os comandos são os seguintes:

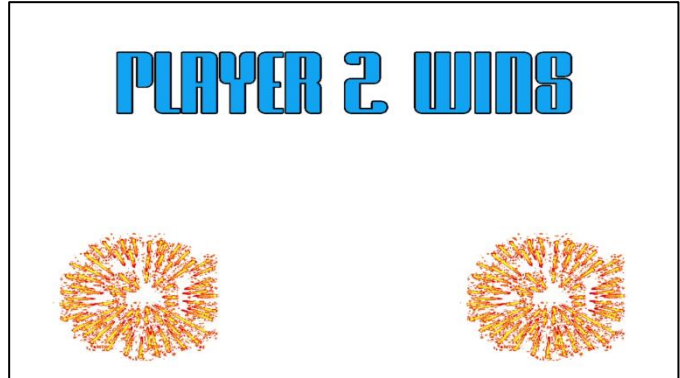
- W para o jogador 1 (lado esquerdo) mover a raquete para cima.
- S para o jogador 1 mover a raquete para baixo.
- ↑ (seta de cima) para o jogador 2 (lado direito) mover a raquete para cima.
- ↓ (seta de baixo) para o jogador 2 mover a raquete para baixo.



Mais uma vez, existe o modo de dia e o modo de noite.

## Winner

Quando um dos jogadores atinge uma pontuação de 6 pontos, um novo ecrã é apresentado. Este ecrã tem uma animação de foguetes que vai repetindo e uma frase a dar os parabéns ao jogador vencedor. Para sair deste ecrã e terminar o jogo basta premir a tecla ESC.



# Estado do projeto

Dispositivo	Funcionalidade	Interrupções
Timer	Controlo de frame-rate e animações	Sim
Keyboard	Controlo das raquetes durante o jogo e atalho para sair	Sim
Mouse	Navegação no menu e seleção de opções	Sim
Graphics card	Display do jogo e determinar colisões	Não
RTC	Determinar hora atual para saber se é dia ou noite	Não

## Timer

O timer é utilizado para controlar os frames do jogo, atualizando a bola e as raquetes com uma frequência de 30Hz.

Para além disso, este é também utilizado para atualizar o sprite de fogo-de-artifício que é apresentado no final do jogo com uma frequência de 5Hz.

## Keyboard

O keyboard é utilizado para mover as raquetes dos dois jogadores de forma independente, estas movem-se desde o momento em que a tecla para determinado efeito é premida até ao momento em que esta é largada.

Este é também utilizado para sair do jogo caso o jogador pretenda, premindo a tecla ESC.

## Mouse

O mouse é utilizado durante o menu para controlar a seta e, quando esta estiver em cima dos botões do menu, é possível premir o botão esquerdo do rato para realizar uma de duas ações, podendo escolher entre iniciar o jogo("Play") ou sair deste("Exit").

## Graphics card

Neste projeto a placa gráfica opera no modo Indexed (0x105) com uma resolução de 1024x768.

Foi utilizado o método de double-buffering para impedir que artefactos fossem apresentados e para tornar o movimento das raquetes e da bola suave e uniforme.

Todas as imagens apresentadas no jogo (exceto os fundos branco e preto) foram feitas a partir de xpm, sendo que a maioria foi deles criados por nós (bola, raquetes, botões e letras do menu) utilizando o Photoshop e o Gimp e outros encontrados na internet de

forma gratuita (fogo de artifício). A fonte utilizada nas letras é chamada “PINGPONG” e foi encontrada de forma gratuita na internet.

Devido á funcionalidade do jogo de ter um modo de dia e de noite, foram criados o dobro dos xpm's pois é necessário um para o modo de dia e outro para o modo de noite. Para isto bastou ir ver a paleta de cores e selecionar uma cor diferente para os caracteres do xpm.

Durante o menu, para impedir que as letras e as imagens de fundo fossem apagadas quando o rato passasse por cima, foi necessário implementar uma função que comparasse a posição do rato com a posição da imagem. Caso o rato passar por cima da imagem, esta é resposta no frame seguinte.

No jogo, foi também necessária uma abordagem semelhante para impedir que a pontuação e as linhas do centro fossem apagadas.

Ainda no jogo, foi necessário criar um algoritmo que controlasse o movimento da bola e a sua atualização sempre que esta colidisse com algo como as paredes do campo e as raquetes. Este algoritmo utiliza também comparações entre a posição absoluta da bola e do objeto com que esta colide.

Para criar transparência nos xpm é utilizada a cor 20. Para este efeito, a função que desenha os pixéis foi alterada para não desenhá-los nada quando recebe esta cor.

## RTC

O Real Time Clock é utilizado para determinar a hora atual do dia. Esta é depois usada para saber se é dia (entre as 8h e as 19h) ou noite. Consoante isto, o modo do jogo vai ser diferente.

Ainda que as funções de interrupções do RTC não tenham sido utilizadas neste projeto, estas foram implementadas na mesma.

# Organização e estrutura do código

## Game

Este modulo é responsável pelo controlo do ciclo de interrupções e por toda a logica do jogo. Para controlar os diferentes estados do jogo foi criada uma State Machine que é atualizada sempre que o estado do jogo muda.

É também aqui se encontram as funções responsáveis pelo movimento e colisões da bola, movimento das raquetes e movimento do rato no menu.

## Timer

Contém as funções que configuram e lidam com as interrupções do timer.

O código foi desenvolvido durante o lab2.

## Keyboard

Contém as funções que configuram e lidam com as interrupções do keyboard.

O código foi desenvolvido durante o lab3.

## Mouse

Contém as funções que configuram e lidam com as interrupções do mouse.

O código foi desenvolvido durante o lab4.

## Video

Contém as funções que configuram o atualizam a placa gráfica. Para alem disso foram depois também adicionadas funções relacionadas com o double buffering utilizado no projeto.

O código foi desenvolvido durante o lab5.

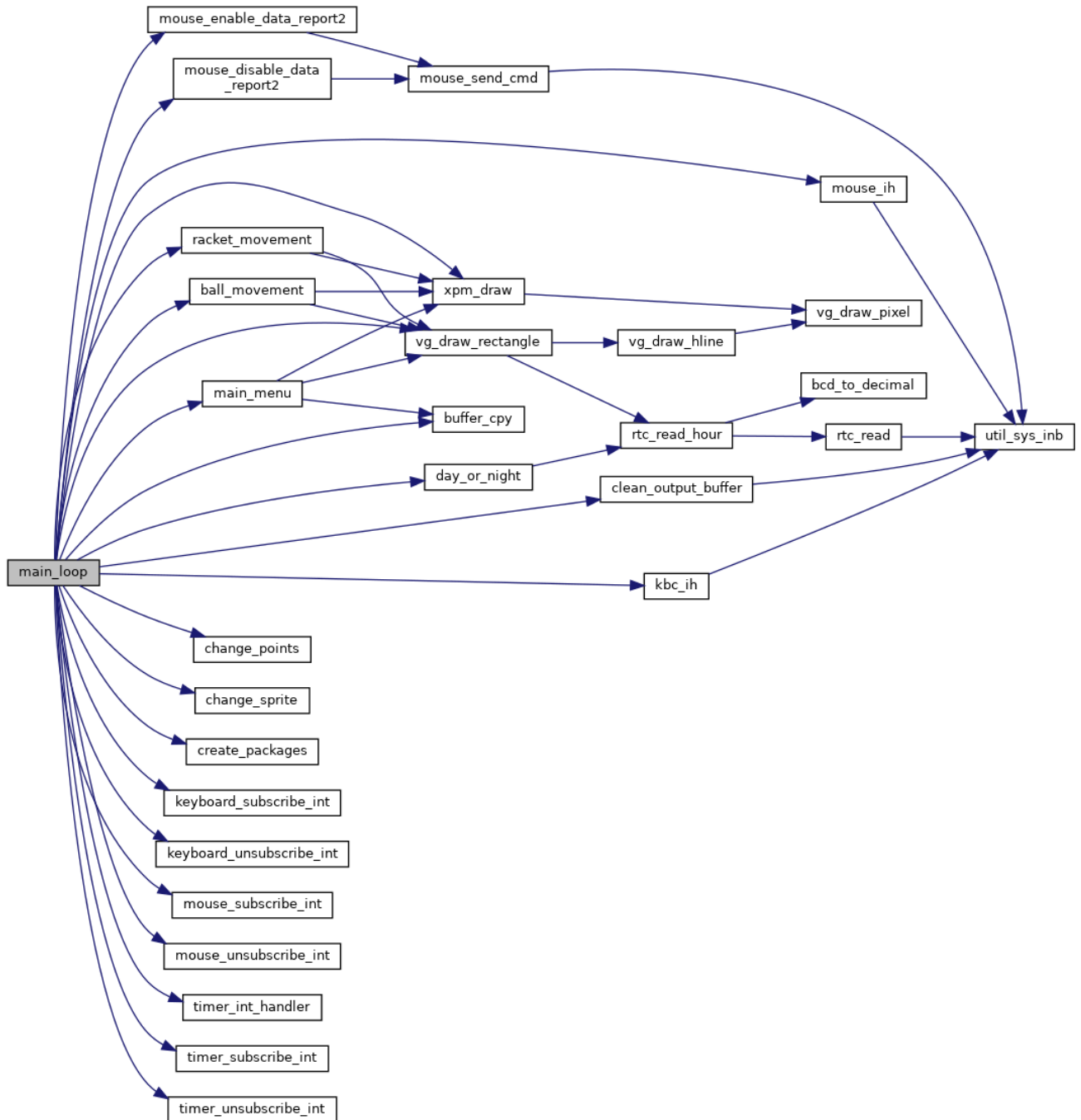
## Proj

Modulo que inicializa e termina o modo gráfico e que chamada a função principal do jogo.

A base deste ficheiro foi dada pelos professores de LCOM e foi adaptada por nós.

**Contribuição final:** Frederico Lopes e Luís Lucas (50% / 50%)

## Function call graph





## Detalhes de implementação

O jogo inteiro acontece dentro do mesmo ciclo while que se encontra no modulo game. Dentro deste ciclo esta implementada uma State Machine que permite um único Drivers Recieve e que controla os interrupts de forma diferente, consoante o estado atual do jogo.

São criadas diversas variáveis globais que são inicializadas no início do jogo e que simplificam muito a atualização destes valores.

Para tornar a organização do código mais simples, foram criadas structs para os xpm's que contêm parâmetros de velocidade, posição e o xpm em si.

Durante o menu, as letras e as opções são todas xpm's diferentes o que permite que caso o rato passe por cima de um dos xpm, apenas este tem de ser desenhado outra vez em vez de o ecrã inteiro.

Para o jogo em si foram criadas duas funções que controlam a bola e as raquetes. O xpm das raquetes é atualizado sempre que algum dos botões que move as raquetes é premido, isto evita uma constante atualização do xpm que iria estar sempre na mesma posição. Por outro lado, a bola e a sua posição são atualizados em todos os frames.

Para as colisões, é tido em conta as posições absolutas dos objetos em questão e a direção da bola é atualizada sempre que isto acontece.

Para tornar a experiência de jogo mais interessante, foram implementados alguns detalhes que achamos importante referir:

- sempre que a bola colide com uma raquete, a velocidade desta aumenta o que evita o jogo de continuar indefinidamente.
- A direção com que a bola começa no início de cada ronda é aleatória, e pode ser em ambos os sentidos (em direção ao jogador 1 ou 2)
- Se a raquete se estiver a mover quando colidir com a bola, a direção da bola é ligeiramente desviada para a direção que a raquete se estava a mover. Isto evita que o ângulo da bola seja sempre o mesmo o que tornaria o jogo repetitivo e torna a física do jogo muito mais realista.

## Conclusões

Apesar deste ter sido provavelmente o projeto mais trabalhoso e difícil realizado neste curso até agora, achamos também que este foi o mais gratificante de fazer. Achamos que esta unidade curricular serve para nos dar uma boa intuição do que acontece nos periféricos e que isso é sem dúvida uma mais valia para o nosso futuro.

Por outro lado, ambos sentimos que no início da unidade curricular é feita uma “abordagem de choque” que torna um pouco difícil conseguir entrar no modo de funcionamento da disciplina, principalmente porque a grande maioria de nós nunca tinha tido qualquer contacto com estes assuntos. Achamos também que por vezes o suporte escrito dos labs não está explicado da melhor forma e pode se tornar um pouco confuso.

Por fim, ambos gostávamos também que as aulas teóricas funcionassem de outra forma pois acreditamos que por vezes estas se podem tornar um pouco redundantes. Visto que a grande parte da matéria já está explicada no suporte escrito, achamos que, quando possível, as aulas teóricas deviam servir para dar uma perspetiva um pouco diferente da que é dada no suporte escrito.

Concluindo, ambos estamos certos de que esta cadeira é uma mais valia no nosso futuro e que a avaliação desta é certamente positiva mas, ainda assim, há vários aspetos que podem ser melhorados.