

Faculdade de Engenharia da Universidade do Porto



CANDY MAZE

Projeto de LCOM grupo T12G02

Manuel Alves (up201906910@up.pt)

Lara Cunha (up202108876@up.pt)

Luís Alves (up202108727@up.pt)

Ana Carneiro (up202008569@up.pt)

Índice

Introdução

1.Instruções de utilização

1.1 Menu Principal

1.1.1 Help Menu

1.2 Mapa

1.3 Pause Menu

2. Dispositivos

2.1 Timer

2.2 Keyboard

2.3 Mouse

2.4 Graphics Card

3. Detalhes de implementação

Introdução

No nosso projeto decidimos realizar um jogo que consiste num labirinto com vários obstáculos. O principal objetivo do nosso jogo é apanhar todos os doces presentes no ecrã e no final do jogo chegar à zonal final.

Durante o jogo o jogador irá ultrapassar vários obstáculos e perder-se no nosso labirinto em busca de todos os doces.

O nome Candy Maze vem do facto do jogo ter como tema principal o mundo do jogo. Esse tema está bastante presente tanto no nosso menu, como no monstro que representa o usuário do jogo e como não podia faltar nos vários doces que o mesmo irá encontrar.

1. Instruções de Utilização

1.1 Menu Principal

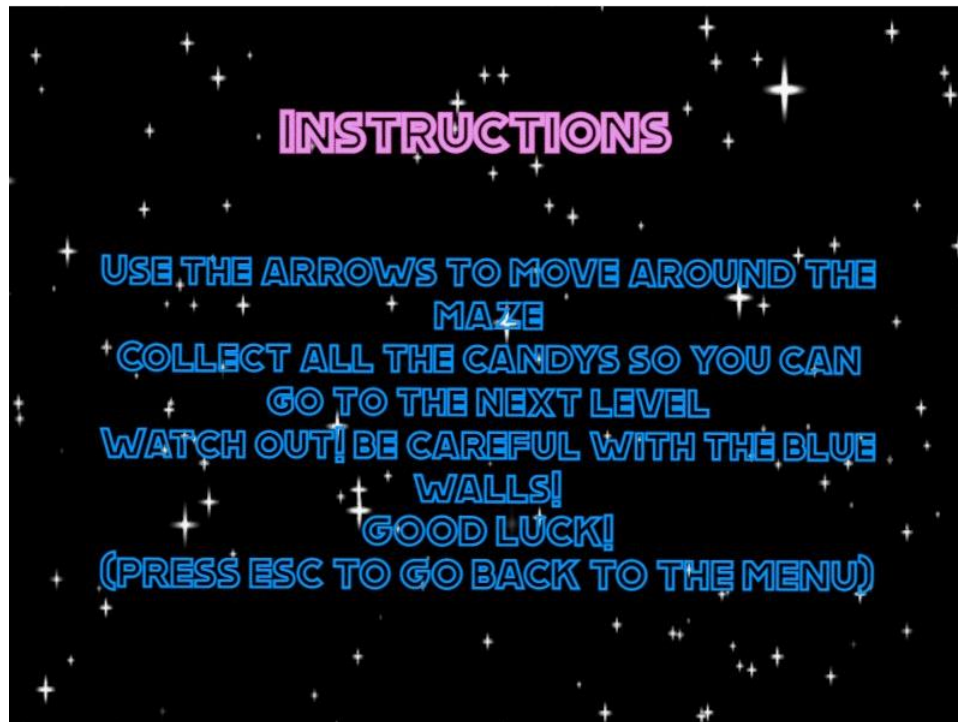


Assim que entramos no jogo podemos observar o menu principal. Este consiste num conjunto de botões denominados por play, help e exit.

Utilizando o rato o utilizador do nosso jogo pode navegar pelo nosso menu e é também através deste que ele pode clicar em cada um dos botões.

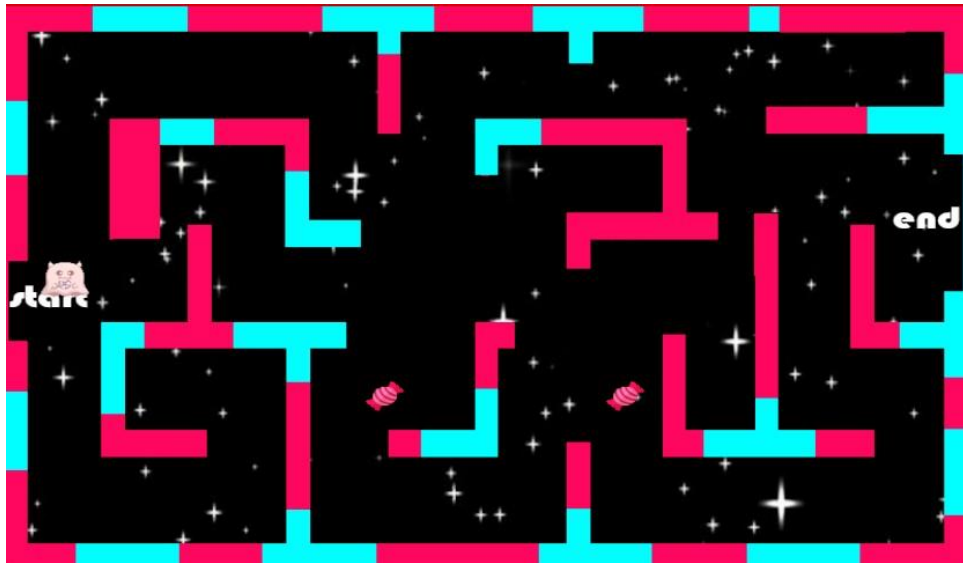
Ao clicar no botão Exit o utilizador sai do jogo. Se clicar com o rato no help, o utilizador será direcionado para uma secção onde temos uma breve explicação do nosso jogo. E claro se clicar no play será direcionado para o primeiro nível do jogo.

1.1.1 Help Menu



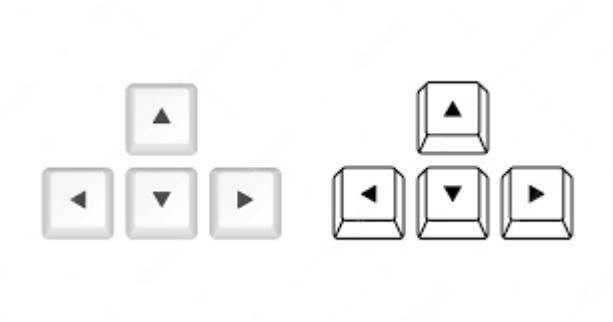
No menu help o utilizador tem uma breve explicação de como jogar o nosso jogo.

1.2 Mapa



Assim que clicamos com o rato no play, o utilizador é direcionado para o nosso primeiro nível. Este nível é bastante simples tendo como principal objetivo o jogador habituar-se ao nosso jogo. Nomeadamente, à maneira de apanhar os doces espalhados no mapa.

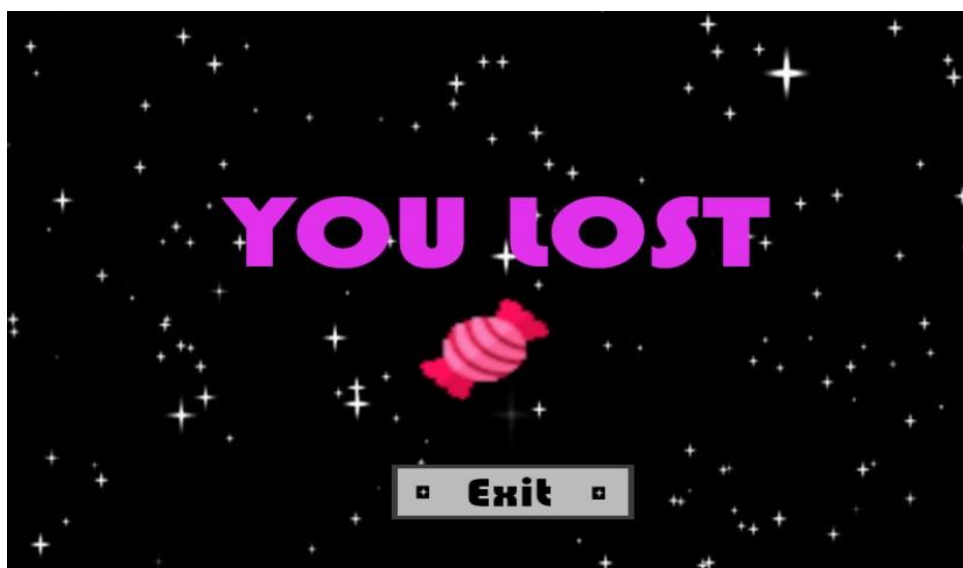
Para apanhar os vários doces espalhados no mapa o jogador necessita apenas de aproximar-se suficientemente deles de forma a colidir com eles. Assim que existe essa colisão o doce é imediatamente apagado do mapa.



O jogador deverá mover-se através das setas do teclado do computador.



O jogador irá navegar pelo labirinto até chegar à zona final, passando ao próximo nível se tiver apanhado todos os doces presentes no mapa e perdendo caso contrário.



1.3 Pause Menu



Durante o jogo o utilizador pode clicar no botão ESC para pôr em pausa o jogo. Nesta página o utilizador terá a oportunidade de continuar o jogo no mesmo ponto onde tinha terminado ou se preferir, sair do jogo e ir para o menu principal.

2. Dispositivos

2.1 Timer

Utilizamos as funções desenvolvidas no lab2 correspondente ao timer para controlar o frame-rate, atualizando constantemente o ecrã, e para gerir todos os eventos durante o jogo, tais como a deteção de colisões com paredes

ou doces e a gestão do movimento do jogador e de quando ele chega à posição final. Para além disso, é utilizado no menu para destacar os botões quando o utilizador passa o cursor em cima dos mesmos.

2.2 Keyboard

Utilizamos as funções desenvolvidas no lab3 correspondente ao keyboard para mover o jogador dependendo da tecla pressionada. Para isto, são utilizadas as arrow keys e os seus respetivos make ou break codes. Para além disso, a tecla ESC pode ser pressionada durante o jogo para o pausar, ou quando se está no menu de ajuda para voltar ao menu principal.

2.3 Mouse

Utilizamos as funções desenvolvidas no lab4 correspondente ao mouse para navegar nos menus e verificar se algum botão foi premido.

2.4 Graphics Card

Utilizamos as funções desenvolvidas no lab5 correspondente ao graphics card, com o modo 0x115 e resolução 800x600. Neste projeto usamos imagens em formato xpm, o que nos permitiu criar o nosso jogador, os vários doces. Para construir as várias páginas, os mapas e os menus usamos um programa para editar imagens.

Com o objetivo de resolver o problema com as colisões, foram criados vários mapas um para cada nível. Sendo, que para criamos uma condição que o monstro não podia passar por cima das cores do nosso labirinto. Assim como para outros objetos como os doces.

Para as imagens ficarem transparentes utilizamos a cor “None”, no modo XPM_8_8_8_8.

3.Detalhes da implementação

Durante a realização do jogo implementamos duas máquinas de estados que tinham como objetivo ter acesso ao estado do jogo. Para que tal acontecesse

chamámos vários Interrupt Handlers dependendo do estado do jogo, o que facilitou o uso de alguns dispositivos em diferentes ocasiões.

Para construir os menus editamos imagens para formar os nossos backgrounds e depois acrescentamos elementos como botões. No nosso projeto temos vários, o botão de play, help, exit e continue. Nas colisões do cursor com os botões usamos a posição relativa e verificamos se a posição está dentro das dimensões do objeto. Isto permitiu-nos criar animações nos botões como quando passamos com o rato por cima deles.

Para além dos ficheiros relativos aos dispositivos, possuímos um main.c, um game.c que possui o GameMainLoop e as interrupções que permitem a execução do jogo, um menu.c que desenha todos os menus e controla os botões neles presentes, um button.c que cria os botões, um level.c que cria e desenha os diferentes níveis e os doces neles presentes e um player.c que desenha o jogador e controla todos os seus movimentos.

Para terminar, houve algumas funcionalidades descritas na especificação inicial do projeto que não conseguimos implementar, maioritariamente por falta de tempo. Entre estas temos o temporizador que dava um certo tempo ao jogador para terminar um nível, perdendo se não o conseguisse fazer antes do mesmo chegar a 0 e os inimigos que andavam pelo mapa e também causavam o jogador perder caso com eles colidisse.