**Laboratório de Computadores**

**Especificação de projeto**



**“DAEDALUS' MAZE”**

**T2G03**

**Nuno Miguel Cardoso Lopes de Freitas**

**José Aleixo Peralta da Cruz**

# Introdução

Para projeto final da cadeira de Laboratório de Computadores (LCOM), o nosso grupo decidiu desenvolver um videojogo. O jogo terá por base um predicado no qual o utilizador é colocado dentro de um labirinto e tem de tentar descobrir a saída, encontrando desafios pelo caminho.

O título "Daedalus' Maze" deve-se ao grego Daedalus que, segundo a mitologia grega, construiu um labirinto complexo para conter o Minotauro.

O labirinto é visualizado pelo jogador a partir de uma perspetiva aérea. No início, apenas verá a sua posição inicial. O labirinto vai-se revelando à medida que o jogador progride no mapa. O utilizador não perderá visão do caminho percorrido anteriormente.

Para enriquecer o jogo, serão colocadas armadilhas ao longo do percurso, às quais o jogador terá de escapar para prosseguir a sua jornada. Esses desafios incluem puzzles, adivinhas ou tarefas que vão sendo cada vez mais elaboradas, à medida que se aproxima a saída do labirinto.

O jogador ganha o jogo se chegar ao final do labirinto. A pontuação dependerá do tempo que demorou a fazê-lo.

Um modo "versus" também será implementado, no qual dois jogadores em computadores diferentes tentarão completar o mesmo labirinto mais depressa que o outro.

# Dispositivos

O plano do nosso grupo é incluir neste trabalho todos os dispositivos que nos foram introduzidos. A tarefa de cada dispositivo é descrita de seguida.

Timer

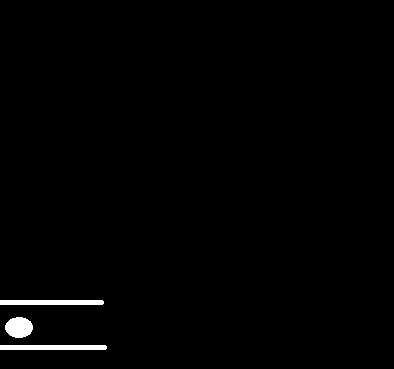
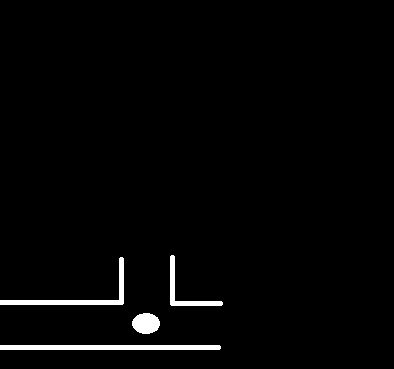
Controlará o tempo que o jogador passou a tomar decisões, que influenciará a pontuação que recebe. Também permitirá implementar animações.

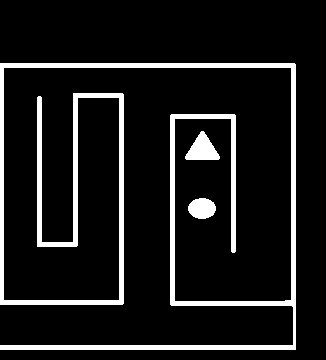
Teclado

Será usado para permitir que o jogador controle a sua personagem e resolva os desafios que lhe vão sendo colocados. Também servirá para seleção de menus. A sua implementação será através de interrupções.

Rato

Em conjunto com o teclado vai ser utilizado para resolver desafios colocados ao jogador. Por exemplo, um desafio em que o utilizador terá de movimentar o rato com um certo padrão para poder prosseguir. Em princípio a sua utilização não será muito complexa, pelo que será utilizado em modo de interrupções.

Modo gráfico



De modo ao utilizador poder visualizar o labirinto e a sua posição neste, aparecerá um “mapa” do labirinto visto a partir de cima, podendo ser observados neste as paredes do labirinto o símbolo do jogador e símbolos específicos para os diferentes desafios encontrados ao longo do jogo. O mapa não aparecerá completo logo de início, sendo completado à medida que o jogador progride neste. Um esquema básico de como desejamos que ele seja visualizado pode ser observado acima.

Caso seja possível, criaremos *non-playable characters* (NPC) que interajam com o jogador, e também para elas criaremos *sprites*.

Real Time Counter

Será gerada uma tabela de pontuações mais altas no jogo, que terá a data em que cada pontuação foi atingida. Essa data será gerada pelo RTC.

Serial Port

Permitirá a troca de informações entre computadores no modo "versus". Uma função verificará a cada intervalo de tempo se um dos jogadores já acabou o jogo. Se acabou é transmitida essa informação para o computador do oponente, terminando o jogo.

# Programação em Assembly

Em princípio serão usados pequenos segmentos de código em Assembly, que sejam de fácil elaboração e compensem em termos de complexidade temporal. Um que será certamente usado é aquele que codifica a função que permite ler uma tecla pressionada no teclado, desenvolvido durante as aulas.

# Lista de módulos

Numa fase inicial, planeamos elaborar um módulo para cada dispositivo e um módulo para cada desafio. Além destes, criaremos módulos particulares para os seguintes:

* Labirinto e funções relacionadas com a visualização pelo utilizador.
* *Sprites* e respetivas funções de animação de NPCs.
* Elementos fixos da interface, como o mini-mapa e a pontuação.
* Possíveis elementos de áudio a incluir.
* Funções auxiliares.
* Função que interliga tudo (o *main()*).

# Plano de desenvolvimento

A nossa abordagem será *bottom-up*, pelo que começaremos por elaborar as funções básicas de cada módulo, construindo a partir daí as funções que darão estrutura ao jogo.

Primeiramente pretendemos desenhar um labirinto e desenvolver as funções associadas à sua visualização pelo jogador para que tenhamos pelo menos uma base com que trabalhar. Depois trabalhamos nos controlos introduzidos pelo jogador, que afetam o percurso da sua personagem ao longo do percurso. Esta parte só é dada por terminada estiver bem assente (sem grandes bugs) e tentaremos terminá-la a tempo da demonstração.

De seguida desenvolveremos apenas 1 ou 2 desafios (incluindo animações de NPC), os respetivos menus de interação e as funções que controlam e guardam as pontuações obtidas (já utilizando o RTC).

O passo seguinte é criar um menu principal para o jogo e inserir todos os elementos fixos no jogo (nome da personagem, pontuação, ...).

Tendo a versão *single-player* estabilizada passaremos à elaboração de uma ligação por Serial Port, para permitir o modo *versus*, que não deverá ser muito complicada, visto que teremos de passar uma única variável.

Com tudo isto completo, caso ainda tenhamos tempo, adicionaremos extras como o visualização do labirinto em primeira pessoa, músicas de fundo, um segundo modo de jogo em que objetivo é guiar o sprite do jogador com o rato sem tocar nas paredes, entre outros elementos.

No fim, esperamos ter um jogo intuitivo, diferente e *bug-free*.