

LICENCIATURA EM ENGENHARIA INFORMÁTICA E DE COMPUTADORES

**Projeto de
Introdução à Arquitetura de Computadores**

TRON

2014 / 2015

INSTITUTO SUPERIOR TÉCNICO

Índice

1. Objetivo	3
2. Descrição e Implementação do Jogo	3
2.1. Início do Jogo.....	3
2.2. Espaço de Jogo	3
2.3. Representação da Partícula	4
2.6. Movimento da Partícula	4
2.7. Níveis de Jogo	4
2.8. Fim do Jogo.....	4
2.9. LEDS	5
2.10. Display de 7 Segmentos	5
2.11. LCD	5
2.12. Versões avançadas	5
3. Plano de Desenvolvimento	6
3.1. Desenvolvimento do trabalho.....	6
3.2. Faseamento da codificação	6
4. Plano de Entrega	7

1. Objetivo

O projeto consiste no desenvolvimento de uma versão do jogo TRON em que o objetivo é bloquear o avanço do adversário. O jogo tem 2 jogadores e decorre na janela de texto do P3. Na janela de texto cada jogador deverá conduzir a sua partícula escolhendo mudanças de direção de modo a limitar o espaço ao adversário e a evitar colisões com os limites e o seu rasto. Neste documento são descritos os detalhes de funcionamento pretendidos para o jogo.

O jogo será programado usando a linguagem *assembly* do P3. O desenvolvimento e teste do programa serão realizados usando o simulador do P3 (p3sim), sendo usados os diversos recursos disponibilizados, de que se destacam: a janela de texto, os *displays* de 7 segmentos, os LEDs, o *display* LCD e os interruptores. O projeto deverá depois ser demonstrado, no laboratório, na placa com a versão hardware do P3.

2. Descrição e Implementação do Jogo

2.1. Início do Jogo

Quando arranca, o programa deverá escrever, centrado respetivamente nas linhas 12 e 13, as mensagens “Bem-vindo ao TRON” e “Pressione I1 para começar”. O jogo deve iniciar-se carregando no botão de interrupção I1.

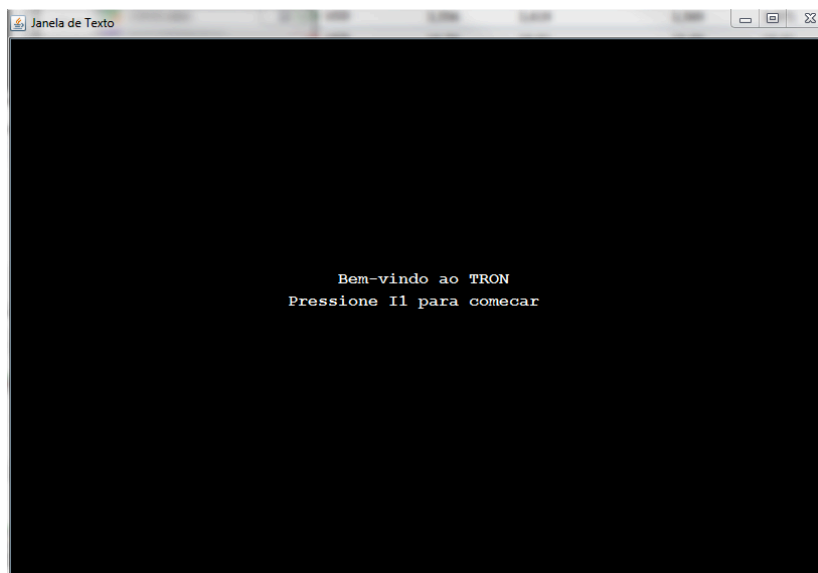


Fig. 1: Início de Jogo

2.2. Espaço de Jogo

O jogo irá desenrolar-se na Janela de Texto do simulador do P3. Esta janela corresponde a uma matriz de 80 colunas por 24 linhas. Deve ser definida através de contornos uma área de jogo de 48 colunas por 20 linhas (sem contar com os contornos da moldura) centrada na janela de texto.



Fig. 2: Espaço de Jogo

2.3. Representação da Partícula

Cada jogador deverá ter um símbolo distinto associado ao seu percurso, por exemplo, '#' e 'X'. Os jogadores devem inicialmente partir das posições (8,10) e (40,10) do espaço de jogo.

2.6. Movimento da Partícula

O movimento das partículas deverá ser controlado por dois botões de interrupção. Para a partícula inicialmente à esquerda deverá ser utilizado o botão I0 e IB, e para a partícula da direita os botões I7 e I9, para definirem respectivamente a viragem à esquerda e à direita.

2.7. Níveis de Jogo

Devem ser implementados 5 níveis de jogo. Cada nível de jogo tem associado uma determinada temporização que condiciona a velocidade do movimento das partículas. As temporizações a considerar em cada nível são 0.7s, 0.5s, 0.3s, 0.2s e 0.1s. O nível de jogo muda aos 10s, 20s, 40s e 60s de cada jogo.

2.8. Fim do Jogo

O jogo termina quando a partícula de um jogador embater no limite do espaço de jogo ou no rasto deixado por qualquer dos jogadores. O programa deverá então escrever, centrado respetivamente nas linhas 12 e 13, as mensagens "Jogo terminou" e "Prima o interruptor I1 para recomencar". Deverá ainda ser atualizado o tempo máximo e as pontuações dos jogadores. O jogo deve reiniciar-se carregando no botão de interrupção I1.

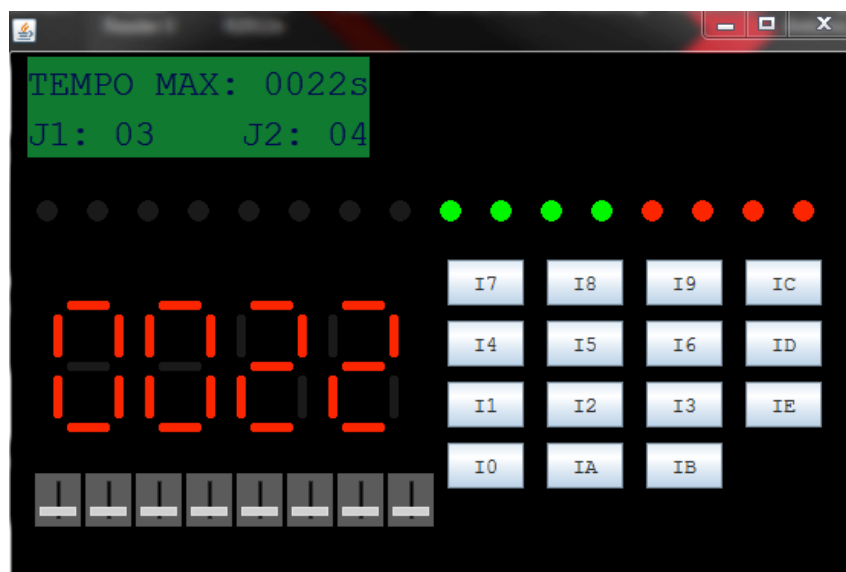


Fig. 3: Janela da Placa

2.9. Leds

Os LEDs devem acender, da direita para a esquerda, em grupos de 4, à medida que os jogadores passam para um novo nível de jogo. No início de cada jogo os leds estão todos apagados.

2.10. Display de 7 Segmentos

No display de 7 segmentos deve ser representado o tempo de jogo em segundos.

2.11. LCD

No LCD deve ser registado o tempo máximo e o número de vitórias de cada jogador (valores obtidos em todos os jogos jogados até então). Por tempo máximo entende-se a duração máxima dos jogos em segundos. Os valores devem ser mostrados como indicado na Figura 3.

2.12. Versões avançadas

Esta versão pode ser estendida de duas formas, independentes entre elas:

- Permitir alterar a dimensão do espaço de jogo. Se o interruptor 0 estiver para baixo o número de colunas deve ser 48 e se estiver para cima o número de colunas deve ser 32. Se o interruptor 1 estiver para baixo o número de linhas deve ser 20 e se estiver para cima o número de colunas deve ser 15.
- Permitir fazer uma pausa com recurso aos interruptores. Se o interruptor 7 estiver para cima o jogo deve ficar em pausa.

Os grupos são livres de introduzirem quaisquer funcionalidades adicionais que valorizem o trabalho.

3. Plano de Desenvolvimento

3.1. Desenvolvimento do trabalho

Sugere-se o seguinte plano de desenvolvimento:

1. Desenhe o fluxograma que descreve cada um dos procedimentos da aplicação, com especial atenção à relação entre o fluxo do programa principal e as várias rotinas de tratamento de interrupção. Estes fluxogramas e a lógica funcional do programa principal deverão ser apresentados na 1ª aula de projeto.
2. Para o conjunto de procedimentos que definiu, identifique claramente as entradas, as saídas e os registos modificados na sua execução.
3. Programe e teste minuciosamente cada uma das rotinas que efetuam a interface com os dispositivos de entrada (interruptores) e os dispositivos de saída (janela de texto, LEDs, display 7 segmentos e LCD), com especial atenção à passagem de parâmetros entre estas rotinas e o programa principal.
4. Configure o temporizador disponibilizado pelo simulador e associe o vector de interrupção respectivo com a rotina a executar periodicamente. Configure o ciclo de jogo para que este seja sincronizado pela mudança de valor de uma dada variável, modificada pela rotina de tratamento da interrupção do temporizador.
5. Realize a ligação entre os vários procedimentos, de forma a obter o comportamento desejado e especificado.
6. Comente e indente devidamente o código desenvolvido. Inclua nos comentários referências aos fluxogramas que irá entregar para auxiliar a leitura e compreensão do programa.

3.2. Faseamento da codificação

Não deve tentar codificar todo o programa de uma só vez. Implemente as várias funcionalidades do programa de uma forma faseada e efetue os testes necessários para verificar o seu correto funcionamento. Não prossiga para a implementação de funcionalidades mais avançadas sem ter garantido que as que lhe servem de base estão corretamente implementadas.

Estando o sistema a funcionar corretamente, pode incluir eventuais funções opcionais que entretanto tenha desenvolvido.

4. Plano de Entrega

Semana de 10 a 14 de Novembro, na aula de laboratório	
Entrega: (em papel)	<ul style="list-style-type: none"> Fluxogramas da aplicação e dos principais procedimentos que lhe servem de base (mesmo que ainda não estejam implementados), com especial atenção à relação entre o fluxo do programa principal, os vários módulos funcionais e as rotinas de tratamento de interrupção.
Demonstração:	<ul style="list-style-type: none"> O programa a apresentar deverá fazer o desenho do cenário de jogo e a colocação das partículas na posição inicial.
Dia 5 de Dezembro, até às 17h00, na Sala de Estudo do DEI	
Entrega: (em papel)	<ul style="list-style-type: none"> Código desenvolvido devidamente comentado e indentedado (impresso frente e verso a duas páginas por face).
Dia 5 de Dezembro, até às 23h59, no Fénix	
Entrega: (electrónica)	<p>A entrega final deverá ser submetida num ficheiro zip com o nome no formato tAaBBgC.zip, em que: A é o dia da semana (2 a 6); BB é a hora de início (basta a hora, com 2 dígitos); C é o número do grupo. Este ficheiro deverá conter:</p> <ul style="list-style-type: none"> Breve relatório em PDF, com a descrição do projeto realizado, organização do programa e explicação dos aspectos mais relevantes da implementação. Na conclusão deverá ser feito um balanço do que foi realizado, com indicação dos aspectos nos quais o projeto tenha divergido do enunciado base (funcionalidades adicionais implementadas, funcionalidades não implementadas, outras variações ou divergências, etc.). Fluxogramas finais da aplicação e dos principais procedimentos que lhe servem de base. Código, quer ficheiro fonte, quer um PDF gerado com a aplicação p3print fornecida na página da cadeira.
Semana de 8 a 12 de Dezembro, na aula de laboratório	
Demonstração:	<ul style="list-style-type: none"> Funcionamento do jogo concebido.

As duas entregas em papel devem ser feitas num envelope identificado com o dia e hora do turno e número do grupo.

O calendário das discussões será acordado com o docente do turno respectivo. A discussão terá lugar, preferencialmente, na semana de 15 a 19 de Dezembro.