

Projeto de IAC

Foi proposta a realização do projeto que consiste na criação do jogo do dinossauro, semelhante ao que se pode encontrar na página do Google, em Assembly no P4.

Estrutura do código:

O código inicia-se com a definição das constantes que irão ser utilizadas no terminal, nas interrupções, no display de 7 segmentos e no temporizador.

No início define-se a máscara das interrupções e guarda-se no endereço 7FF8h a altura da posição do dinossauro, de seguida, há um loop (ciclo_inicio) que mantém o jogo parado até ser pressionado o botão '0'.

Quando '0' é pressionado, é colocado todo o display a '0', é definido o cursor para FFFFh, para limpar o terminal, e são definidas as variáveis a ser usadas nas funções geracato e atualizajogo, depois, entra num loop que tem como função escrever o chão do campo de jogo com o carácter '═'.

Terminado o loop, passa-se para a função TIMER, que inicializa o temporizador e escreve no display o valor do temporizador em decimal, que corresponde à pontuação obtida no jogo.

Após o incremento do temporizador, inicializa-se a função OUTPUT, que escreve no terminal todos os valores da tabela "Campo" na linha acima da correspondente ao chão (linha 2B), sendo o valor '1' correspondente ao cato com 1 de altura representado por '||' e o valor '2' correspondente ao cato com 2 de altura representado por dois '||'.

Seguidamente, após se ter escrito a tabela toda, passa-se para a função pos_dino, onde se escreve a posição do dinossauro, representado pelo carácter '♥'. Nesta função é verificado se foi realizada a interrupção do botão 'UP', se sim, inicia-se o salto do dinossauro, em que a função atualiza a posição do dinossauro para cima uma vez por cada atualização do terreno, até ao endereço 280Ah, a partir de onde começa a descer até voltar à posição inicial. A função verifica ainda se houve colisão do dinossauro com um cato, se sim passa para a função game_over, que apaga o terminal e faz reset à tabela na memória que representa o campo de jogo, se não, passa para a função geracato que gera um valor (1 ou 2) para o próximo cato a aparecer no terminal.

Depois, inicia-se a função atualizajogo, que vai deslocar todos os membros da tabela na memória 1 posição para a esquerda e, na posição mais à direita, é adicionado o valor gerado pela função geracato.

No fim, volta-se à label LOOP, que realiza o aumento do temporizador e repete-se todo o processo.

Desenho da aplicação:

Foi feita a decisão de colocar o chão por baixo da linha onde é colocado o campo do jogo por questões de estética, assim como os símbolos que representam tanto os catos como o dinossauro.

Foi colocado no endereço 7FF8h a altura do dinossauro pretendida, para depois poder ser usado na função `pos_dino`, que usa e altera o valor que foi colocado nesse endereço de memória para poder guardar o valor da altura do dinossauro e não ser alterado quando se volta a iniciar todo o processo do código.

Decidimos colocar duas alturas diferentes nos catos para ficar mais próximo do jogo original.

Foi escolhido também criar as constantes `KEY0` e `KEYUP` para verificar se os respetivos botões foram pressionados. `KEY0` passa para 1 quando é pressionada e volta a 0 quando se dá o 'GAME OVER'. `KEYUP` passa para 1 quando é pressionada e volta a 0 quando o dinossauro chega à altura máxima do salto.

NOTA:

Está também neste ficheiro, além do que contém o projeto final, o correspondente ao temporizador que conta em decimal.

Projeto realizado por:

Daniel Gomes - 99195

Gonalo Carmo - 99228