

- **Introdução**

O projeto de IAC 2016/17 consiste na programação de uma versão minimalista do jogo de *Arcade R-Type*. Neste jogo, o utilizador controla uma nave que se pode movimentar em 4 direções, cada uma associada a uma interrupção (INT0 para baixo, INT1 para cima, INT2 para a esquerda e INT3 para a direita), tentando atingir o máximo número de asteróides com tiros que a nave pode disparar (INT4 para disparar). Durante o jogo, vão sendo gerados obstáculos (na forma de asteróides e de buracos negros) consecutivamente e de forma aleatória na última coluna da janela de texto. Os buracos negros não poderão ser destruídos, enquanto a destruição dos asteróides irá aumentar o score. O jogo acaba quando a nave colide com algum dos obstáculos ou com o próprio tiro (usando a INTE para reiniciar o jogo a qualquer momento durante o mesmo ou para o começar), sendo que depois de acabar salta para um *Game Over* screen com a pontuação disponível, sendo esta igual ao número de asteroides destruídos. O jogo é regido por um Clock (INTF), sendo esta a que dá ao jogo o seu ritmo, controlando o movimento e geração de obstáculos.

- **Desenvolvimento**

Para tornar este jogo funcional, tivemos que implementar diversas funções, referentes a cada uma das *features* do jogo, por vezes intercalando-as umas com as outras. Assim, passo a descrever todas as funcionalidades do programa.

Inicialização

Escreve uma mensagem inicial, preparando o utilizador para começar o jogo. Quando esta é premida, a janela é limpa e escreve-se a janela de texto com cardinais a indicar os limites. A inicialização é comum ao restart e limpa tanto os obstáculos como todas as posições de memória relativas ao programa. A inicialização escreve a nave na sua posição inicial, sendo que a nave é representada por “\”, “(”, “/”, “>”.

Movimento

Funções referentes ao movimento para cada uma das 4 direções, sendo estas chamadas no ciclo Main cada vez que a interrupção associada ao seu movimento é gerada. Antes do movimento ser efetuado, estas funções verificam a colisão com qualquer objeto antes do movimento ser efetuado, gerando uma Flag de Game Over se a Colisão for ocorrer.

Tiro

Esta rotina cria um tiro na posição da frente do canhão, desde que um tiro não exista. O tiro é incrementado até colidir ou até ao chegar ao limite da janela de texto. É representado por “_”

Asteróides

Os asteróides são um dos dois tipos de obstáculos, sendo estes possíveis de destruir e representados por “*”. Cada vez que um asteróide colide com um tiro, os dois desaparecem e a posição é incrementada no display, mostrando-a em hexadecimal.

Buracos Negros

Os buracos negros representam o segundo tipo de obstáculos, sendo estes impossíveis de destruir (só quando chegam ao limite da janela de texto), e ao colidir com um tiro só o tiro é que é destruído. São representados por “O”.

Display de Sete Segmentos

O display de sete segmentos mostra a pontuação, sendo que esta é updated cada vez que um asteróide é destruído, somando 1 à pontuação.

LCD

O LCD mostra a posição atual do canhão da nave, sendo atualizado cada vez que a nave é movimentada.

LEDS

Os LEDS rapidamente acende cada vez que um asteróide é destruído, sendo apagados no ciclo main do jogo.

Ciclo Main

O ciclo main serve como uma base de qual todas as funções/funcionalidades do jogo são chamadas, retornando à sua posição neste ciclo. Para tar, cada interrupção Incrementa ou realiza uma instrução MOV a uma flag.

Game Over

A funcionalidade Game Over é representada por uma flag no ciclo Main, que quando acionada inicia a sequência de Restart ao jogo, necessitando da INTE para o recomeçar.

Colisões Tiro/Obstáculo

Para verificar a colisão de um tiro com um obstáculo, o ciclo de colisão compara a posição atual do tiro com a posição de todos os obstáculos que atualmente existem, tanto asteróides como de Buracos Negros. Se forem iguais, realiza uma colisão baseada no tipo de obstáculo.

Colisões Nave/Obstáculo/Tira

Ao movimentar a nave, vão ser verificadas as posições de todas as peças da nave e da posição de todos os obstáculo e do tiro. Se forem iguais, incrementa-se a flag Game Over que acaba o jogo.

- **Conclusão**

A nossa versão de jogo final ficou jogável, com todas as funcionalidades básicas e algumas avançadas. O código podia estar mais curto, mas à falta de tempo para cortar rotinas ficou como está. Ainda tentamos implementar a função níveis, que usaria uma INT5 para passar a um nível muito mais difícil que o habitual, mas tal não foi possível. Assim, o jogo corre quase com a ausência completa de bugs, excepto um que é 1 buraco negro parar quando se reinicia o jogo.