

# **Introdução à Arquitetura de Computadores**

## **Relatório do Projeto “GORILLAS”**

### **Grupo 78**

**Inara Parbato - 91110**

**Ricardo Calvão - 92545**

- **Descrição do programa**

O programa baseia-se no jogo **GORILLAS**. Trata-se de uma versão simplificada do jogo, com apenas um jogador e um simples contador de pontos. O jogo não acaba, simplesmente aumentando os pontos obtidos sempre que o gorila adversário é atingido. Sempre que um ponto é obtido os gorilas mudam para uma posição aleatória nas suas respetivas bases.

- **Variáveis globais e constantes importantes**

Para facilitar a escrita e a leitura do código definimos as constantes que representam os endereços na memória das interrupções.

Definimos também variáveis globais para poder simular a comunicação entre as diferentes funções do projeto. Variáveis como ‘ANG’, ‘V’, e ‘T’, que representam, respetivamente o ângulo, a velocidade e o instante no tempo que vão ser alterados por certas funções e que depois são utilizadas para o cálculo da trajetória do projeto.

Várias cadeias de caracteres são também guardadas em variáveis, para reduzir a repetição do código e a quantidade de linhas do programa no seu total.

- **Interrupções**

As interrupções são inicializadas no início do programa. As interrupções apenas são ativadas quando são necessárias. O botão A é utilizado para iniciar o jogo e é logo de seguida desativado. Cada um dos botões, de 0 a 9, são ativados logo depois do botão A ser desativado passando para a parte inicial do jogo quando se pede o ângulo e a velocidade. Ao premir um botão uma função é chamada que guarda um número, de 0 a 9

respetivamente, na variável global **BOTAO**. O timer é utilizado para representar os frames das diferentes animações do programa a cada 10 milissegundos.

- **Modo de funcionamento**

No início do programa imprime-se um gráfico a dizer '**GORILLAS**' no ecrã juntamente com uma mensagem a dizer para carregar no botão 'A' para começar o jogo. Estas cadeias de caracteres são escritas com a ajuda da função **ESCREVE\_STRING**, que escreve uma cadeia de caracteres na posição dada como argumento. Esta função é chamada sempre que se escreve uma cadeia de caracteres no ecrã.

Depois cria-se um loop que espera até o botão 'A' ser premido, isto sendo verificado através da variável **START**, alterada pela interrupção chamada pelo botão. Enquanto o botão não é premido incrementa-se o valor da variável **SEED**, que vai servir como um valor pseudoaleatório para ser utilizado mais à frente no programa. Este valor estabiliza-se após o botão 'A' ser premido e o programa procede.

Aqui inicia-se a parte **MAIN** do programa. Definem-se as coordenadas onde os gorilas se vão situar na consola com a função **DEF\_COORDS**, que gera duas coordenadas aleatórias para os dois gorilas e guarda-as nas variáveis **X0** e **X1**. Os números aleatórios são obtidos com a função **RAND** que pega no valor da **SEED** armazenado e o altera com base no valor da última velocidade inserida no programa (guardado em **V**). Esta função gera um número aleatório de 16 bits. Na função **DEF\_COORDS** é utilizado este número para gerar duas coordenadas entre certos valores definidos. Como apenas a posição no eixo horizontal dos gorilas varia, apenas um número de 8 bits é suficiente para a representar. O byte mais significativo e o menos significativo do número aleatório representam as diferentes coordenadas.

A função **DESENHA\_JOGO** é chamada, que por si vai chamar as funções **DESENHA\_GORILA** duas vezes (uma para cada gorila, utilizando as coordenadas calculadas anteriormente), e o **DESENHA\_PONTOS** que vai desenhar os prédios, que apenas servem como bases para os gorilas.

Na função **LER\_ANG\_V** limpam-se os valores de **V** e **ANG** e ativam-se as interrupções dos botões. Primeiramente a função espera que se prima num dos botões para guardar o valor e escreve-lo no ecrã e depois espera-se por um segundo valor que vai ser adicionado ao

*ANG* multiplicado por 10. Apenas dois valores são permitidos na introdução do ângulo. A mesma lógica é aplicada a introdução do valor da velocidade.

Depois reinicializa-se o valor da variável *ACERTO*, que vai ser utilizada para eventualmente verificar se a banana acertou num gorila ou não.

A função ***ATIRAR*** é chamada. Aqui faz-se uma animação do gorila a levantar a mão para atirar a banana e inicializa-se o timer. O timer, a cada 10 milissegundos, chama a função *BANANA* (apenas se *ACERTO* estiver no seu valor inicial) e incrementa *T*.

A função *BANANA* utiliza a variável *POS\_ANTERIOR*, que representa coordenadas na consola, para apagar o caractere que se encontra nessa posição. Depois chama-se a função “***func\_pos***” (função criada na primeira parte do projeto) para calcular a posição da banana no tempo *T* indicado. Verifica-se se essa posição se encontra fora do ecrã ou se acertou num gorila ou num prédio. Dependendo do caso a função passa para *BANANA\_CAIU* ou *BANANA\_ACERTO*. Se acertou ***ACERTO*** é incrementado duas vezes. Algumas das variáveis são subsequentemente inicializadas. Se a banana se encontra ainda no ar a função continua o ciclo.

Se ***ACERTO*** tiver um valor de 1 (quer dizer que a banana acertou num gorila) chama-se a função ***ACERTO\_ANIM***, que incrementa os pontos e faz a animação do gorila que foi acertado. Este frame da animação é mantido no ecrã utilizando o timer.

Se a banana não acertou faz-se um *branch* para *OUTRA\_VEZ* que reinicia o jogo sem calcular novas coordenadas para os gorilas. Se a banana acertou o jogo recomeça, desta vez calculando novas coordenadas.