**Relatório de Sistemas Operativos 2**

**Meta 1**

**Daniel Simões | 21210390| 14 de Maio de 2017**

# **Breve introdução**

# Com este trabalho, pretende-se que os alunos apliquem boas práticas de programação orientada a objetos e/ou usar corretamente a linguagem c, usando as restrições arquiteturais requeridas.

Pretende-se assim, que seja implementado, na plataforma Win32, um jogo de habilidade de condução de serpentes, semelhante ao clássico jogo ‘’Snake’’.

## **Memória partilhada**

A memória partilhada será usada para conter o mapa que será escrito pelo servidor e acedido/lido pelos vários clientes locais. Por sua vez, cada cliente irá escrever a posição para a qual se pretende movimentar, e após validação por parte do servidor, será reconstruído o mapa, caso necessário.

## **Estruturas de dados**

Snake -> Para já, contem um id de cada cobra, comprimento, posição X e Y da cabeca, velocidade e se está viva.

Game -> Estrutura usada para guardar tudo sobre o jogo. Contem as várias Snakes e também o mapa do jogo.

Nota: O mapa do jogo é representado num array bidimensional TCHAR onde cada posição tem um caracter que corresponde a determinada coisa (item, cobra, chao, parede, etc.)

## **Funções da DLL**

writeMapInMemory() -> Função que o servidor irá chamar para escrever o mapa em memória, para ser depois lida pelos vários clientes (Ainda não funcional)

readMapInMemory() -> Função que o cliente irá chamar para ler o mapa em memória (Ainda não funcional)

generateEventNames() -> Função interna da DLL. Usada para gerar os nomes a associar a cada evento.

## **Aspetos de sincronização**

Neste momento ainda não foram usados grandes mecanismos de sincronização. No entanto, já foi percebido que será necessário usar mecanismos de sincronização para aceder às zonas de memória partilhada. De momento estão a ser usados eventos e semáforos.