



Instituto Superior de Engenharia de Coimbra
Licenciatura em Eng.Informática

Relatório do Trabalho Prático Sistemas Operativos 2 2021/22

Meta 1

Diogo Pascoal 2018019825

Nuno Honório 2019126457

INTRODUÇÃO

O trabalho prático de Sistemas Operativos 2 consiste na implementação do “Jogo dos Tubos” numa versão multiutilizador. O jogo consiste em conduzir um caudal de água de um ponto de origem para um ponto de destino de uma área retangular no monitor. Para esse efeito, o jogador terá de utilizar as peças/tubos mais adequados ao percurso pretendido.

Este trabalho é concretizado em linguagem C, com o objetivo de perceber e utilizar a WinApi32.

Este sistema é, nesta primeira meta, composto apenas do Servidor, que processa a lógica do jogo, e o monitor que observa o próprio jogo.

ESTRUTURAS

Tabuleiro

```
typedef struct {
    int tam;
    TCHAR tab[20][20];
    int startCel;
    int destCel;
    int agua;
    int aguay;
    int aguax;
    int tempAgua;
}Tabuleiro;
```

Esta estrutura representa o nosso campo de jogo, indicando o tamanho do tabuleiro, a célula de início e fim, a velocidade da água e posição, e guarda a representação do mesmo tabuleiro.

Memória Circular

```
typedef struct {
    int posE;
    int posL;
    int nProdutores;
    int win;
    Tabuleiro tabMem;
    CelulaBuffer buffer[TAM_BUFFER];
}MemoriaCircular;
```

Estrutura usada para a memória circular aplicada na memória partilhada entre os processos.

CelulaBuffer

```
typedef struct {  
    int id;  
    TCHAR comando[3][20];  
} CelulaBuffer;
```

Representa uma célula da memória circular onde é armazenado um comando e os seus possíveis argumentos.

DadosThread

```
typedef struct {  
    MemoriaCircular* memPar; //ponteiro para a memoria partilhada  
    HANDLE hSemEscrita; //handle para o semaforo que controla as escritas (controla quantas posicoes estao vazias)  
    HANDLE hSemLeitura; //handle para o semaforo que controla as leituras (controla quantas posicoes estao preenchidas)  
    HANDLE hMutex;  
    HANDLE hEventoMapa;  
    int terminar; // 1 para sair, 0 em caso contrário  
    int id;  
} DadosThreads;
```

Representa as informações que as threads em geral necessitam para o seu funcionamento, desde aos handles para os semáforos que são necessárias em várias threads, assim como acesso à memória partilhada.

COMUNICAÇÃO

Os programas comunicam entre si através do uso da memória partilhada. Através desta ferramenta, o Servidor envia o tabuleiro para o monitor de modo a este o poder apresentar.

Da mesma maneira, para o Servidor receber comandos, o Monitor procede a escrever na memória partilhada os comandos recebidos pelo utilizador e envia-os para o Servidor para serem processados.

São utilizados Semáforos para garantir sincronização entre os processos neste processo de uso da memória partilhada, assim como Eventos para notificar o monitor de uma atualização no servidor, de modo a ir atualizar o seu tabuleiro para apresentá-lo ao utilizador. Ao iniciar, os semáforos também permitem garantir que o servidor está iniciado pois o servidor é o único que pode criar os mesmos.

Existe um semáforo para controlar as escritas do servidor para transmitir o tabuleiro para os monitores, e outro para o servidor obter comandos vindos dos monitores.

Existe também um Mutex criado no Servidor de modo a garantir que este é único e não existem várias instâncias deste programa.

USO DO SISTEMA

Para utilizar o sistema, basta iniciar em primeiro o servidor. O servidor deve ser iniciado com a seguinte linha de comando:

Servidor.exe <Tamanho Mapa> <Velocidade Água>

Caso o tamanho do mapa e da velocidade da água não seja indicado, o servidor irá recorrer ao registry para obter estes valores. Caso não estejam definidos no registry, o programa irá tentar criar estas chaves, inserindo valores default que depois poderão ser alterados pelo utilizador manualmente nos registos do Windows.

Caso o utilizador indique um valor inválido, o programa também irá recorrer ao registry para ter um valor válido para iniciar o jogo.

Após a obtenção dos valores necessários para iniciar, o servidor irá esperar que o utilizador indique para iniciar através do comando “começar”. Isto irá iniciar o jogo e os monitores serão capazes de ver o progresso do jogo.

Nota: Se iniciar os monitores antes de iniciar o servidor, os mesmos irão apresentar uma mensagem de erro indicando que não foi possível se conectar ao servidor.