

## Relatório do Trabalho Prático de Sistemas Operativos Simulador para Offloading de Tarefas no Edge

Rui Pedro Capelas Santos 2020225542  
Tomás Bernardo Martins Dias 2020215701

### **Breve síntese sobre o funcionamento do programa:**

Quando o programa é iniciado é criado o processo “SystemManager” sendo feita a leitura do ficheiro “config.txt”, que contem informações das restantes funcionalidades do programa, especialmente sobre o número de EdgeServers presentes.

Este é responsável para criar a memória partilhada, message queue, e o named pipe que servira para comunicar com os EdgeServers. Sendo também responsável por criar tanto o processo Monitor, que controla o número de vCPUs dentro de cada Edge Server e o seu nível de performance, o processo TaskManager responsável por criar e enviar as tarefas para os Edge Server através do pipe anteriormente criado. Por último é criado o processo MaintenanceManager que iria comunicar através da message queue e colocar os Servidores em manutenção.

O processo TaskManager como referido anteriormente cria os EdgeServers (e um unnamed pipe associado a cada Server) e os seus dois vCpus (duas threads) como também duas threads, o scheduler e o dispatcher, que são responsáveis pela recepção e envio das tarefas respetivamente, que são enviadas pelos MobileNodes que são criados num ficheiro.c à parte.

O programa ao receber o sinal SIGTSTP através do SystemManager ou uma mensagem “STATS” no TaskManager imprimirá as estatísticas do sistema de offloading e terminará ao receber o sinal SIGINT através do SystemManager ou uma mensagem “EXIT” no Task Manager.

Todas as informações sobre as ações realizadas ao longo da execução do programa estão presentes num ficheiro nomeado “log.txt”.

### **Shared Memory (Memória Partilhada):**

De modo que os processos tenham acesso a variáveis que são alteradas por outros processos, implementou-se uma zona de memória partilhada.

### **Named Pipe:**

As tarefas geradas pelos mobile nodes são enviadas pelo named pipe para o TaskManager: Este apresenta uma capacidade máxima definida no ficheiro “config.txt” e sobre ele atuam o dispatcher e o scheduler.

### **Unnamed Pipes:**

O número de unnamed pipes depende do número de EdgeServer contidos no ficheiro “config.txt”, e estes servem para comunicar entre o TaskManager e os EdgeServer disponíveis.

### **Message Queue:**

A comunicação entre o MaintenanceManager e o EdgeServers é feita através de uma message queue. Sendo enviado o tempo de manutenção do servidor pelo MaintenanceManager e recebida a resposta por parte do EdgeServer da disponibilidade de realizar manutenção (caso se encontre ou não a realizar tarefas).

### Sinais:

O processo SystemManager pode receber dois sinais: SIGINT(CRTL+C) e SIGTSTP(CRTL+Z). Sendo os restantes ignorados com recurso a máscaras.

Quando o SystemManager recebe o sinal SIGINT(CTRL+C), este envia aos seus processos filho um sinal SIGTERM que eles vão captar e lidar de forma ao programa encerrar de forma seguro, por exemplo esperando que as tarefas dos EdgeServer terminem e só depois encerrar o programa.

Quando o SystemManager recebe o sinal SIGTSTP(CRTL+Z), este imprime as estatísticas sobre o funcionamento do sistema de offloading.

### Mecanismos de sincronização:

Recorremos ao uso de semáforos named POSIX, para controlar a concorrência na escrita no ficheiro “log.txt” e na Shared Memory. Para além disso utilizamos variáveis de condição juntamente com mutexs evitando assim esperas ativas e problemas de concorrência respetivamente.

### Horas de trabalho:

Rui Pedro Capelas Santos- 64 horas

Tomás Bernardo Martins Dias- 27 horas

### Esquema do projeto:

