

Licenciatura em Engenharia Informática

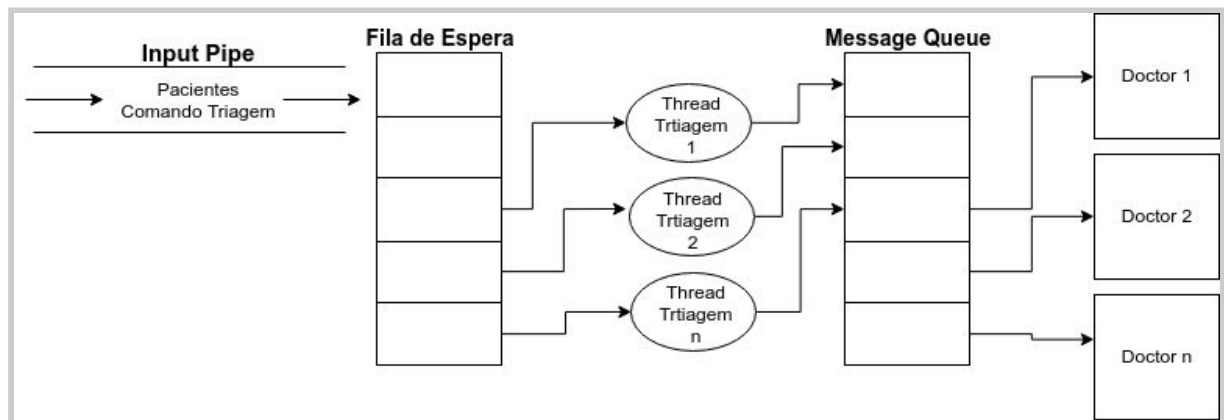
2º Ano

1º Semestre

## Relatório do Projeto Prático de Sistemas Operativos

Rui Pedro Costa Linhares, 2015227420

Tiago Guilherme Lopes Gomes, 2015245562



Esquema do funcionamento do Sistema

## Descrição do Funcionamento Global do Sistema

Aquando do início do servidor, é lido o ficheiro config.txt e são criadas as threads e os processos doutor de acordo com a informação do ficheiro. É também criada a memória partilhada para o armazenamento de estatísticas. É criado e aberto o named pipe para leitura, inicializada a fila de espera e a message queue e por fim inicializados os mecanismos de sincronização necessários (mutexes e semáforos posix).

### Named Pipe

Foi definido um protocolo para a receção de mensagens pelo named pipe:

Pacientes únicos: nome\_triagemtime\_doctortime\_prioridade (**ex. joao\_10\_10\_1**);

Grupo de pacientes: número\_triagemtime\_doctortime\_prioridade (**ex. 8\_10\_10\_1**)

Neste caso, são gerados números aleatórios que servirão para identificar cada um dos pacientes do grupo;

Após a receção dos pacientes, estes são colocados na fila de espera (uma lista ligada FIFO).

Comando triagem: triagem\_nthreads (**ex. triagem\_10**)

Neste caso, as threads atuais dão exit (pthread\_exit) e o server dá join das mesmas. São criadas novas threads consoante o número enviado através do comando.

## Triagem

A triagem é realizada pelas threads. Cada thread começa por ir à fila de espera e retira de lá o paciente que está lá há mais tempo. Espera o tempo definido na estrutura paciente. As estatísticas são atualizadas e o paciente é enviado para a message queue.

## Message Queue

As mensagens enviadas pela Message Queue são a estrutura paciente em que o long mytype é representado pela prioridade **(1 a 3)**. Assim, as mensagens cuja prioridade é 1 serão as primeiras.

## Processos Doutor

Os processos doutor são responsáveis por receber os pacientes pela message queue no seu turno, esperar o tempo correspondente e atualizar as estatísticas. Têm um turno definido pelo ficheiro config.txt. Se estão a tratar de pacientes e o seu turno acaba, terminam o tratamento desse cliente e o processo é terminado, sendo que outro é criado pelo processo gerador de processos.

Existe também o processo do doutor extra. Este é criado quando o tamanho da message queue excede o tamanho máximo definido pelo ficheiro config.

## Sincronização

Foram usados os seguintes mecanismos de sincronização:

**Mutexes** (pthread\_mutex) para a sincronização das threads;

**Semáforos Posix** para a sincronização na escrita dos dados na memória partilhada e acesso à queue;

Manual de Utilizador:

Compilar com gcc -Wall -pthread main.c -o main

## Esforço

**Rui Linhares - 20 horas**

**Tiago Gomes - 20 horas**