

SISTEMAS OPERATIVOS

2019/2020

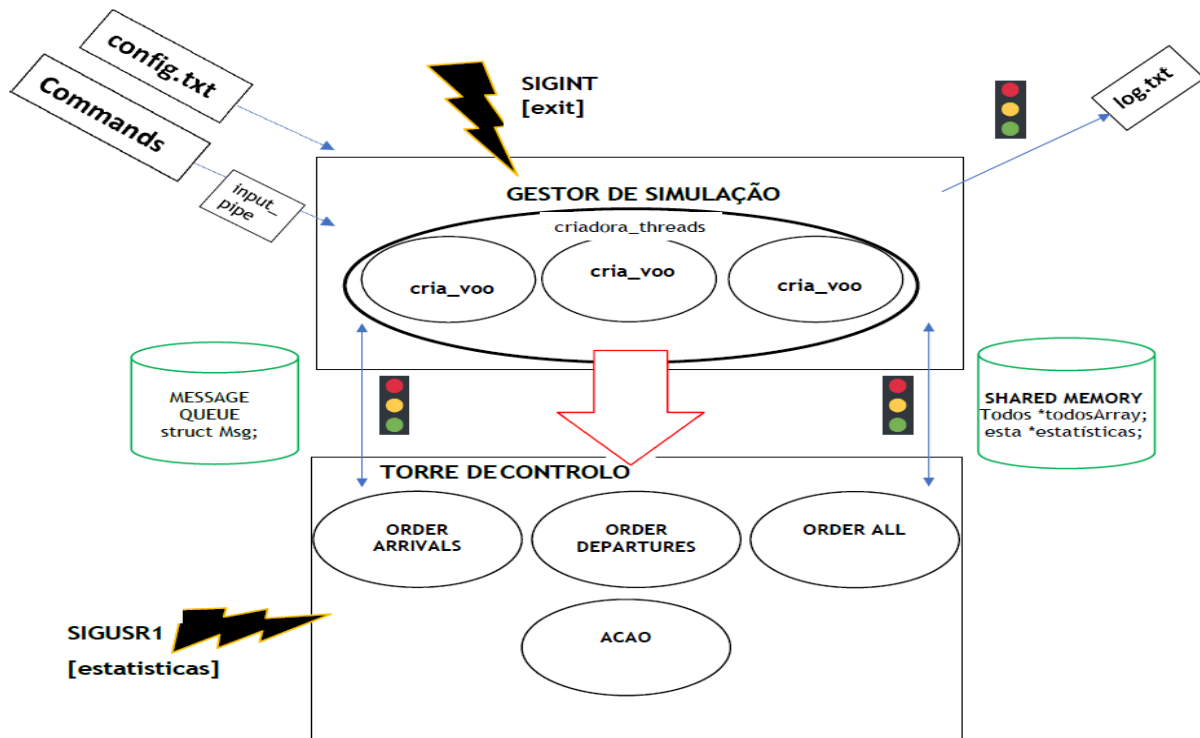
LICENCIATURA ENGENHARIA INFORMÁTICA 2º ANO – 1º SEMESTRE

RODRIGO SOBRAL

2018298209

RAUL NOGUEIRA

2017267634



Descrição geral do programa:

Quando o processo *gestor* arranca, são inicializadas a *shared memory*, *named_pipe*, a *message queue* e os semáforos, as alocações de memória são feitas, o ficheiro *config.txt* é lido e o ficheiro para o registo do histórico de eventos é criado, o *log.txt*. De seguida, bifurcamos o processo e criamos a Torre de Controlo.

SHARED MEMORY:

As estatísticas e o *array todosArray* ficarão armazenados numa zona de memória partilhada. No caso das estatísticas é para que ambos os processos possam incrementar os seus contadores, no caso do *todosArray*, bem, vai ser o a estrutura de dados onde os voos vão ficar organizados por ordem de execução.

SEMÁFOROS:

Foram inseridos semáforos no programa 2 semáforos. Um para limitar o acesso à memória partilhada e o outro para limitar o acesso à escrita no ficheiro log, isto porque são os únicos recursos onde mais que um processo ou thread podem ter acesso simultaneamente.

SINAIS:

Existem dois sinais no programa: o SIGINT que só será aplicado no Gestor de Simulação e o SIGUSR1 que será apenas aplicado na Torre de Controlo. O SIGINT encerrará o programa eliminando toda a informação criada e encerrando a Torre. O SIGUSR1 irá somente imprimir as estatísticas registadas até ao momento.

INPUT PIPE:

A *input_pipe* vai ser a *named_pipe* que vai receber os comandos vindos do terminal. O Gestor apenas aceitará dois tipos de comandos:

ARRIVAL {flight_code} **init:** {initial time} **eta:** {time to runway} **fuel:** {initial fuel}
DEPARTURE {flight_code} **init:** {initial time} **takeoff:** {takeoff time}

MESSAGE QUEUE:

A *message queue* será apenas responsável por transmitir a informação dos voos recebida para a Torre.

LOG.TXT:

O ficheiro *log.txt* será criado no início do programa e será nele que iremos escrever qualquer tipo de ocorrência (comandos bem/mal inseridos, voos criados, holdings, voos aterrados/descolados, etc) e a sua hora.

GESTOR DE SIMULAÇÃO:

Vai estar repetidamente à escuta de comandos na *input_pipe*, verifica-os, insere-os numa lista ligada, ordena-os por init e de seguida e a enviá-los para a Torre de Controlo.

TORRE DE CONTROLO:

Vai ficar infinitamente à escuta de comandos na *Message Queue* até que lhe seja aplicado um sinal vindo do processo pai, o Gestor de Simulação. Ao receber os dados dos voos vindos do Gestor, vai fazer o escalonamento de todos eles tendo sempre em conta os seus ETAs e Fuel's no caso das aterragens e dos seus Takeoffs no caso das descolagens

TEMPO INVESTIDO:

Raul Nogueira : 60h
Rodrigo Sobral: 45h