



# Universidade de Brasília (UnB)

## Departamento de Ciência da Computação

**Disciplina:** Sistemas Operacionais

**Prof:** Alba Melo

**Período:** 02/2023

### Descrição do Trabalho Prático (02/2023)

#### 1. Pré-requisitos

Os alunos deverão conhecer a linguagem de programação C e chamadas de sistema Unix.

#### 2. Descrição

O projeto consiste da confecção de um escalonador de loteria (*lottery scheduling*) para aplicações compostas por múltiplos processos CPU-bound, que duram cerca de 30 segundos.

O processo escalonador de aplicações recebe um arquivo, contendo um nome de processo e uma prioridade por linha e lê o arquivo. Existirão 3 prioridades e (a) processos com prioridade 1 receberão 2 *tickets*, (b) processos com prioridade 2 receberão 3 *tickets* e (c) processos com prioridade 3 receberão 4 *tickets*.

O processo escalonador possui quantum de 6 segundos e os processos serão executados da seguinte maneira: (a) o primeiro processo do arquivo inicia a execução (vai para a fila *ready*) logo que o arquivo é lido; (b) os demais processos iniciam a execução nos próximos 2 segundos (vão para a fila *ready*), e. g., o segundo processo vai para a fila *ready* em 2 segundos, o terceiro processo vai para a fila *ready* em 4 segundos e assim por diante. De maneira assíncrona, o escalonador entra em atividade a cada 6 segundos (quantum), pára a execução do processo corrente, roda o escalonador de loteria e coloca o processo escolhido para rodar.

O escalonador de loteria vai escolher de maneira randômica um *ticket* e o processo que o tiver vai ser executado. O escalonador vai atribuir *tickets* a cada rodada, com base no número de processos na fila *ready* e suas prioridades. Por exemplo, se tivermos 3 processos na fila *ready*, com prioridades 1, 2 e 3, o domínio dos tickets é {0, 1, 2, 3, 4, 5, 6, 7, 8} (2+3+4). Nesse caso, os processos com prioridades 1, 2 e 3 poderiam receber os tickets {2, 5}, {0, 1, 6} e {3, 4, 7, 8}, respectivamente. Para decidir qual processo executar, o escalonador gera um número pseudo-randômico utilizando MOD 9.

Se o grupo preferir, podem ser criados dois processos: um para criar os processos e os colocar na fila *ready* e outro para executar o escalonador. Utilizar somente um processo para essas duas funções também é possível.

#### 3. Output

Ao término da execução, o processo escalonador da aplicação deverá imprimir o *makespan* (tempo de *turnaround*) e o tempo de execução de cada processo. O *makespan* é calculado a partir do momento que o processo for criado até o momento em que o processo termina a execução. O tempo de execução é calculado a partir do momento em que o processo é colocado em execução pela primeira vez até o momento em que o processo termina a execução.

#### 4. Atenção: só podem ser utilizadas as chamadas de sistema de processos discutidas em sala de aula. Não podem ser utilizadas threads ou as chamadas semaphore.