

UNIVERSIDADE FEDERAL DO MARANHÃO  
CURSO DE CIÊNCIA DA COMPUTAÇÃO  
Disciplina: Sistemas Operacionais I  
Prof: Mário Meireles Teixeira  
Aluno: \_\_\_\_\_

Cód: \_\_\_\_\_

6,3  
NL

## 1ª Avaliação

Marque **apenas uma alternativa correta** em cada questão. (3,0)

1. O algoritmo SJF (Shortest Job First):
  - a) Sempre dá prioridade ao processo mais antigo. *x*
  - ☒ b) Pode ser preemptivo ou não-preemptivo. *C*
  - c) É o algoritmo padrão no Linux. *x*
  - d) Garante a execução simultânea de processos. *x*
2. O campo "Program Counter" armazenado no PCB serve para:
  - a) Controlar o uso de memória. *x*
  - b) Armazenar o PID do processo. *x*
  - c) Indicar o próximo processo a ser escalonado. *C*
  - ☒ d) Indicar a próxima instrução a ser executada. *3,0*
3. O algoritmo Round-Robin é mais adequado para:
  - a) Sistemas de tempo real estritos.
  - b) Sistemas batch com cargas longas.
  - ☒ c) Sistemas interativos com resposta rápida. *C*
  - d) Ambientes de execução única.
4. Classifique as afirmativas abaixo como (V) ou (F). (1,0)
  - F* a) Os sistemas operacionais modernos surgiram na década de 1970 com o advento dos computadores pessoais. *F*
  - F* b) No Linux, o comando `cd` é utilizado para listar os arquivos de um diretório. *F* *na lista*
  - F* c) O ciclo de vida de um processo inclui os estados: Novo, Pronto, Executando, Esperando e Finalizado. *F*
  - V* d) O comando `fork()` no Linux é responsável por criar um novo processo que herda as características do processo pai. *V*
  - F* e) O escalonamento FIFO é preemptivo e permite que processos mais curtos interrompam o processo em execução. *F* *nao*
  - F* f) No escalonamento Round-Robin (RR), o tempo de execução de cada processo é dividido em fatias de tempo chamadas de quantum. *time slice* *F*
  - F* g) O algoritmo Shortest Job First (SJF) é sempre a melhor escolha, pois evita qualquer tipo de starvation. *F*
  - V* h) O Process Control Block (PCB) armazena informações importantes, como identificador do processo, ponteiro de pilha e estado atual do processo. *V*

0,8

- F i) No Linux, a chamada de sistema exec cria um novo processo filho enquanto mantém o processo pai em execução.
- V j) No escalonamento com múltiplas filas (MLFQ), processos que consomem mais CPU tendem a ser rebaixados para filas de menor prioridade.

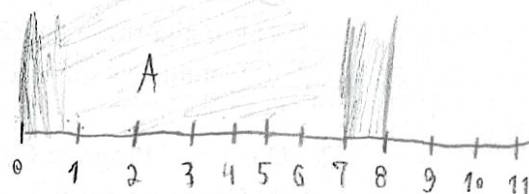
5. Considere o cenário de um sistema com três processos aguardando para serem executados. A tabela abaixo mostra o tempo de chegada e o tempo de execução de cada processo. (3,0)
- a) Qual a ordem de execução dos processos usando os algoritmos FIFO, SJF e STCF (SJF Preemptivo)?
- b) Calcule o tempo de espera médio utilizando os três algoritmos acima.

Considere:

**Tempo de Espera:** É o tempo total que um processo permanece na fila de prontos (esperando para ser executado), excluindo o tempo de execução.

Processos:

Processo	Tempo de Chegada	Tempo de Execução
P1	0	8
P2	1	4
P3	2	2



Resposta:

$$T_{\text{Esp}} = T_{\text{prontos}} - t_{\text{chegada}}$$

Algoritmo	Ordem de execução	Tempo de Espera Médio
FIFO	1111111222233	$(0+7+10)/3 = 5,6$
SJF	1111111332222	$(0+6+9)/3 = 5$
STCF	1233222111111	$(0+0+0)/3 = 0$

2,5

FIFO:  $c_1$  0  $c_2$  1  $c_3$  2 3 4 5 6 7 8 9 10 11 12 13  
 $P_1$  chega  $P_2$  chega  $P_3$  chega  $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_2$   $P_2$   $P_2$   $P_2$   $P_3$   $P_3$   
 $[P_1$   $P_1$   $P_1$

$$T_{P1} = 0 - 0 \rightarrow 0$$

$$T_{P2} = 8 - 1 \rightarrow 7$$

$$T_{P3} = 12 - 2 \rightarrow 10$$

SJF:  $c_1$  0  $c_2$  1  $c_3$  2 3 4 5 6 7 8 9 10 11 12 13  
 $P_1$  chega  $P_2$  chega  $P_3$  chega  $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_3$   $P_3$   $P_2$   $P_2$   $P_2$   $P_2$   
 $[P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_3$   $P_3$   $P_2$   $P_2$   $P_2$   $P_2$

$$T_{P1} = 0 - 0$$

$$T_{P3} = 8 - 2 \rightarrow 6$$

$$T_{P2} = 10 - 1 \rightarrow 9$$

$$\frac{17}{3} \approx 5,6$$

$$\rightarrow \frac{15}{3} = 5$$

STCF: 0 1 2 3 4 5 6 7 8 9 10 11 12 13  
 $P_1$  chega  $P_2$  chega  $P_3$  chega  $P_3$   $P_2$   $P_2$   $P_2$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   
 $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$   $P_1$

$$T_{P1} = 0 - 0$$

$$T_{P2} = 1 - 1 \rightarrow 0$$