



INSTITUTO FEDERAL
Pernambuco

Exercício 05

Disciplina: Sistemas Operacionais.

Curso: Análise e Desenvolvimento de Sistemas.

Turno: Noite.

Professor: Me. José Paulo Lima.

Data de entrega: 18/06/2025.

Aluno (a): _____

Matrícula: _____

1. O que é escalonamento?
2. Quais as condições que fazem uma CPU escalonar?
3. Quais os objetivos do algoritmo de escalonamento?
4. Qual a diferença do escalonamento preemptivo e não preemptivo?
5. Diferencie os termos abaixo que envolve o escalonamento de processos:
 - (a) Processos preemptivos e processos não-preemptivos.
 - (b) *Turnaround*
 - (c) *Throughput*
 - (d) Tempo de espera

6. IF-SC - 2014 - IF-SC - Professor - Informática (Adaptada)

Sobre algoritmos de escalonamento de processos:

A. Múltiplas Filas

— Nesse algoritmo o primeiro processo a chegar será o primeiro a ser executado.

B. Round-Robin

— Nesse algoritmo é definido um quantum (fatia de tempo) para cada processo. Após encerrar a fatia de tempo, o processo escalonado deve ceder o lugar na CPU a outro.

C. Shortes Job first

— Nesse algoritmo são usadas várias filas de processos prontos para executar. Cada processo é colocado em uma fila e cada fila tem uma política de escalonamento.

D. Escalonamento Lotérico

— Nesse algoritmo o sistema distribui bilhetes aos processos, e faz um sorteio cada vez que precisa selecionar um processo para a CPU.

E. First Come First Serverd.

— Nesse algoritmo os processos menores terão prioridade, ou seja, serão executados primeiro.

7. Assinale V para as alternativas verdadeiras e F para as falsas:

- A. ☒ V ☐ F Os sistemas interativos não fazem necessário a utilização de algoritmos de escalonamento preemptivos devido ao seu grande número de usuários.
 - B. ☒ V ☐ F O algoritmo de escalonamento First Come, First served é flexível permitindo a podendo escolher qualquer processo da sua lista para executar na CPU.
 - C. ☒ V ☐ F No algoritmo de escalonamento First Come, First served os processos podem executar quanto tempo quiser na CPU.
 - D. ☒ V ☐ F O algoritmo Shortest Job First permite que tarefas mais curtas possam ser executadas primeiramente antes das tarefas mais longas.
 - E. ☒ V ☐ F O algoritmo Remaining Time Next é similar ao algoritmo Short Job First, ou seja, ao receber um novo processo cujo o tempo de execução é maior do que o tempo restante do processo corrente, o algoritmo suspenderá o processo corrente e executará o novo processo.
8. Cinco processos, de A até E, chegam ao computador ao mesmo tempo. Eles têm seus tempos de processamento estimados em 10, 6, 2, 4 e 8 minutos respectivamente. Suas prioridades (atribuídas externamente) são 3, 5, 2, 1 e 4, respectivamente, sendo 5 o representante da prioridade mais alta. Nenhum dos processos faz I/O. Para cada um dos algoritmos de escalonamento abaixo, determine o tempo médio de turnaround dos processos. Ignore o overhead causado pela troca de contexto.
- (a) Round Robin (fila começa em A, indo em ordem até E ; quantum = 4)
 - (b) Escalonamento com prioridade.
 - (c) FIFO (ordem de execução: A, B, C, D, E)
 - (d) SJF
9. Considere o seguinte conjunto de processos, com o tamanho do tempo de burst de CPU dado em milissegundos:

Processo	Tempo de Serviço	Prioridade
P_1	10	3
P_2	1	1
P_3	2	3
P_4	1	4
P_5	5	2

Considere que os processos chegaram na ordem P_1, P_2, P_3, P_4, P_5 no momento 0.

- (a) Desenhe quatro gráficos que ilustrem a execução desses processos usando FCFS, SJF, prioridade não-preemptiva (um número de prioridade menor significa uma prioridade mais alta) e o escalonamento Round Robin (quantum = 1).
- (b) Qual é o turnaround de cada processo para cada um dos algoritmos de escalonamento no item (a)?
- (c) Qual é o tempo de espera de cada processo para cada um dos algoritmos de escalonamento no item (a)?
- (d) Qual dos escalonamentos no item a resulta no menor tempo de espera médio (em relação a todos os processos)?

10. A tabela abaixo apresenta alguns processos a serem organizados por um algoritmo de escalonamento. Considerando que cada processo tem seu respectivo tempo de criação/chegada, tempo de execução e prioridade, organize estes processos de acordo com os algoritmos não preemptivos abaixo:

Processo	Chegada	Execução	Prioridade
A	3	6	1
B	4	7	3
C	1	3	2
D	8	8	6
E	7	2	4
F	5	4	7
G	2	9	8
H	0	5	5
I	10	1	9

- FIFO (First In First Out) ou FCFS (First Come First Served)
 - SJF (Shortest Job First)
 - Por prioridades
 - Round-Robin com *quantum* igual 3.
 - Represente detalhadamente como ficaria a ordem de execução dos algoritmos SJF e por Prioridades caso eles fossem preemptivos.
 - Calcule o tempo de espera de cada processo em cada algoritmo e indique qual algoritmo teve em média menores tempo de espera.
11. Cinco processos são criados na seguinte ordem: P_1 , P_2 , P_3 , P_4 e P_5 , com os seguintes tempos:

Processo	Tempo de serviço	Prioridade	Tempo de chegada
P_1	13	3	0
P_2	11	4	4
P_3	7	1	5
P_4	8	2	7
P_5	16	5	10

Ilustre a execução dos processos através de um diagrama usando os seguintes esquemas de escalonamento:

- FIFO
 - SJF
 - Prioridade (número de prioridade menor implica prioridade maior)
 - Circular com fatia de tempo = 4 u.t.
 - Mostre os tempos de turnaround individuais por processo e a média resultante.
- Desconsidere E/S ou tempo de escalonamento ou troca de contexto entre processos.