



### Considerações:

1. Trabalho composto de 3 questões valendo 2 pontos extras na Avaliação Parcial 01.
2. Equipes: máximo de 3 alunos.
3. Os códigos devem ser entregues pelo GitHub até a data prevista. Apenas um membro da equipe deve submeter o link do repositório na atividade. Os demais apenas submetem um txt informando o nome do colega que enviou.
4. A equipe deverá apresentar os códigos ao professor em data a ser definida. Nota individual.

### Questão 1:

[1 ponto]

Crie uma simulação utilizando o algoritmo de escalonamento *Round Robin*. Dado  $N$  processos com diferentes *burst time*, analise as métricas *tempo médio de espera*, *tempo médio de retorno* e *vazão* para diferentes valores de *quantum*. Considere:

- Todos os  $N$  processos tem o mesmo tempo de chegada.
- 1 unidade de tempo é gasto para a mudança de contexto do processo.
- *Tempo médio de espera* : Tempo médio que um processo passa no estado de pronto antes de ser executado.
- *Tempo médio de retorno* : Tempo médio que um processo leva para ser concluído após sua chegada ao sistema.
- *Vazão* : Quantidade de processos que são concluídos em um determinado período de tempo.
- Produza gráficos.

Exiba a sequência de execução de cada processo para os diferentes valores de *quantum*. Utilize a linguagem de programação de seu interesse.

### Questão 2:

[1/2 ponto]

No problema do jantar dos filósofos, crie um protocolo que permita uma operação sem impasse. Utilize como base os algoritmos disponíveis em:

<https://docs.oracle.com/cd/E19205-01/820-0619/gepji/index.html>

Não utilize a solução disponível no material citado.

Verifique se ocorre impasse em 1000 execuções.

### Questão 3:

[1/2 ponto]

No problema dos leitores e escritores, realize uma simulação para comparar duas soluções. Na primeira, os escritores são obrigados a esperar para acessar a região crítica sempre que houver leitores, fato que pode levar os escritores a esperar indefinidamente pelo acesso. Na segunda, essa possibilidade de espera indefinida dos escritores é resolvida. Discuta os achados.

Leitura recomendada:

COURTOIS, Pierre-Jacques; HEYMANS, Frans; PARNAS, David Lorge . Concurrent control with “readers” and “writers”. Communications of the ACM, v. 14, n. 10, p. 667-668, 1971.