

| **Disciplina:** Estrutura de Dados I | | |
| --- | --- | --- |
| **Modalidade:**  Presencial |  | **Professor:** André Lucio de Oliveira |

**TRABALHO PRÁTICO (A1)**

Data de Entrega: **01/12/2022**

Possibilidade de fazer a atividade em trio

Pontuação máxima: 10,0 (dez)

Forma de entrega: por e-mail (andre.oliveira@uva.br) + apresentação para o professor em sala (o aluno deverá demonstrar conhecimento sobre o código fonte entregue.

OBS: O campo "Assunto" do e-mail deve ter o seguinte padrão:

* ED1-TRABALHO\_PRATICO\_A2\_NOME\_INTEGRANTES\_GRUPO

Especificação da atividade:

Implementar um escalonador de processos em C, considerando um SO time sharing e o conceito de lista de prioridades com envelhecimento de tarefas.

Os dados de execução deverão ser lidos de um arquivo txt com o seguinte formato:

7

0 34 1

1 10 9

1 4 6

2 78 5

3 5 1

4 9 3

4 10 6

Onde:

* Linha 1 = número total de processos presentes no arquivo
* Nas linhas seguinte serão listados os processos a serem executados:
  + Coluna 1 = Instante de entrada do processo
  + Coluna 2 = tempo total de execução
  + Coluna 3 = Prioridade estática (quanto mais alto o inteiro, maior a prioridade)

Dados de um nó na fila de processos:

* PID (process ID) - gerar randomicamente
* Tempo total de execução - ler do arquivo
* Tempo restante para finalizar a execução - atualizado durante a execução
* Instante de entrada do processo
* Prioridade estática - ler do arquivo txt
* Prioridade dinâmica - atualizado durante a execução

OBS:

* O valor de quantum (tempo máximo que um processo fica na CPU) pode ser uma constante no seu programa.
* Da mesma forma, vocês podem definir o valor “alfa”, que é o fator de incremento da prioridade dinâmica.
* Vocês podem simular (através de sleep) que 2ns, por exemplo, seriam 2 seg.