



**UNIVERSIDADE FEDERAL DO MARANHÃO**  
**BACHARELADO INTERDISCIPLINAR EM CIÊNCIA E TECNOLOGIA**  
**EECP0009 - ARQUITETURA DE COMPUTADORES**

**ALEXSANDRO NOGUEIRA COSTA FILHO**

**JEYSRAELLY ALMONE DA SILVA**

**JOSE NUNES DE SOUSA NETO**

**LETICIA BORGES ASSUNCAO**

**SAMARA SANTOS VIEGAS**

**MANUAL DO USUÁRIO DE ESCALONAMENTO DE PROCESSOS**

**SÃO LUÍS - MA**

**JANEIRO/2025**

ALEXSANDRO NOGUEIRA COSTA FILHO (2022009300)

JEYSRAELLY ALMONE DA SILVA (20240064879)

JOSE NUNES DE SOUSA NETO (2022003263)

LETICIA BORGES ASSUNCAO (2022030028)

SAMARA SANTOS VIEGAS (2022042898)

## **MANUAL DO USUÁRIO DE ESCALONAMENTO DE PROCESSOS**

Documento apresentado como requisito parcial de avaliação da disciplina Arquitetura de Computadores - Turma 01, no curso Bacharelado Interdisciplinar em Ciência e Tecnologia da Universidade Federal do Maranhão.

Orientador: Prof. Me. Luiz Henrique Neves Rodrigues.

SÃO LUÍS - MA

JANEIRO/2025

## SUMÁRIO

<b>Manual do Usuário - Escalonador de Processos.....</b>	<b>4</b>
1. Execução da Versão Python.....	4
Requisitos:.....	4
Como Executar:.....	4
2. Execução da Versão C.....	5
Requisitos:.....	5
Como Compilar e Executar:.....	5
3. Funcionalidades da Interface Gráfica.....	5
Adicionar Processos:.....	5
Remover Processos:.....	6
Escolher o Algoritmo de Escalonamento:.....	6
Visualização dos Resultados:.....	6
4. Dicas de Uso.....	7
5. Solução de Problemas.....	7

# Manual do Usuário - Escalonador de Processos

Este manual explica como executar o programa "Escalonador de Processos" em suas versões Python e C, explorando a interface gráfica e as funcionalidades do sistema. O objetivo é permitir ao usuário adicionar processos, escolher diferentes algoritmos de escalonamento e visualizar os resultados graficamente.

---

## 1. Execução da Versão Python

### Requisitos:

1. Python 3.9 ou superior
2. Bibliotecas adicionais:
  - **tkinter** (para a interface gráfica)
  - **matplotlib** (para os gráficos)

### Como Executar:

Instale o Python e as bibliotecas necessárias com o comando:

```
pip install matplotlib
```

- 1.
2. Salve o código Python fornecido em um arquivo chamado **escalonador.py**.

Abra o terminal ou prompt de comando, navegue até o diretório onde o arquivo está salvo e execute:

```
python escalonador.py
```

- 3.
4. A interface gráfica abrirá automaticamente, permitindo o uso do programa.

**Obs:** Para execução da versão em Python apresentada em sala de aula, o usuário pode baixar o arquivo .rar contido no Drive/Github do projeto e rodar uma versão já executável em seu computador.

---

## 2. Execução da Versão C

### Requisitos:

1. **Compilador C:**
  - GCC (recomendado).
2. **Bibliotecas para GUI:**
  - GTK ou equivalente, dependendo da implementação da interface gráfica.

### Como Compilar e Executar:

1. Salve o código C em um arquivo chamado **escalonador.c**.

Compile o programa com o GCC, incluindo as flags necessárias para as bibliotecas de interface gráfica. Por exemplo, para GTK:

```
gcc escalonador.c -o escalonador `pkg-config --cflags --libs gtk+-3.0`
```

- 2.

### Execute o programa:

```
./escalonador
```

- 3.
4. A interface gráfica abrirá automaticamente.

---

## 3. Funcionalidades da Interface Gráfica

### Adicionar Processos:

1. Insira os dados do processo nos campos correspondentes:
  - **PID:** Identificador único do processo.
  - **Duração:** Tempo total de execução do processo.
  - **Chegada:** Tempo de chegada do processo no sistema.
2. Clique em Adicionar Processo. O processo será listado na tabela de processos.

#### **Remover Processos:**

1. Selecione um processo na lista.
2. Clique em Remover Processo para excluí-lo.

#### **Escolher o Algoritmo de Escalonamento:**

1. Use o menu suspenso para selecionar um dos algoritmos:
  - **FIFO (First In, First Out)**
  - **SJF (Shortest Job First)**
  - **SRTN (Shortest Remaining Time Next)**
2. Clique em Executar Escalonamento para processar os dados.

#### **Visualização dos Resultados:**

1. Tabela de Resultados:
    - A tabela exibe as seguintes informações para cada processo:
      - **PID**
      - **Tempo de Duração**
      - **Tempo de Chegada**
      - **Tempo de Espera**
      - **Turnaround**
      - **Resposta**
  2. Gráfico de Gantt:
    - Após a execução de um algoritmo, um gráfico de Gantt será exibido para ilustrar o escalonamento.
-

## 4. Dicas de Uso

- Antes de iniciar, tenha clareza sobre o algoritmo desejado, pois cada um possui comportamento único.
  - Insira valores válidos (inteiros) nos campos; entradas incorretas serão ignoradas.
  - Salve os resultados ou capture a tela dos gráficos para documentação futura.
- 

## 5. Solução de Problemas

- Erro ao Executar:
  - Certifique-se de que todas as dependências estão instaladas.
  - Verifique se o arquivo está salvo corretamente e se o comando de execução foi digitado corretamente.
- Interface Não Abre (C):

Garanta que as bibliotecas de interface gráfica estejam instaladas. No caso de GTK:

`sudo apt-get install libgtk-3-dev`

○

- Gráfico Não Aparece (Python):

Atualize a biblioteca **matplotlib** com:

`pip install --upgrade matplotlib`

○

---

Com essas instruções, você estará apto a explorar todas as funcionalidades do "Escalonador de Processos".