

# Trabalho de SO

Levi Cícero Arcanjo   Victor Emanuel Almeida

13 de outubro de 2020

# Conteúdo da apresentação

- 1 Introdução
- 2 Item 2 do projeto
- 3 Item 3 do projeto
- 4 Item 4 do projeto

# Uma simples introdução

teste

# Simulação dos algoritmos de escalonamento

## Características do programa:

- Linguagem: C++
- Algoritmos:
  - Round Robin
  - Shortest Job First
- Todos os processos iniciaram juntos  
(Possuem tempo de criação zero)

# Simulação dos algoritmos de escalonamento

## Dados do Process Control Block implementado:

```
/**
 * Class: Represents a process control block
 */
class PCB {
private:
    int PID;
    string name = "";
    State state;
    PCB * next;
    double createdTime = 0;
    double estimatedTime = 0;
    double remainingTime = 0;
```

# Algoritmos Utilizados

## Round Robin:

### Definição

Algoritmo preemptivo que atribui fatias iguais de tempo a cada processo, manipulando todos os processos sem prioridade e em ordem circular.

```
/**  
 * Class: Represents a RR algorithm  
 */  
class RR{  
private:  
    Queue * ready;  
    Queue * terminated;  
    PCB * processRunning;  
    double quantum;
```

# Algoritmos Utilizados

## Shortest Job First:

### Definição

Algoritmo não preemptivo que seleciona para ser executado o processo com o menor tempo de execução.

```
/**  
 * Class: Represents a SJF algorithm  
 */  
class SJF{  
    private:  
        Queue * ready;  
        Queue * terminated;  
        PCB * processRunning;
```

# Problema clássico

- Como problema clássico escolhemos o “Leitores e escritores”



# Problema clássico

- Como problema clássico escolhemos o “Leitores e escritores”
- Então propusemos um algoritmo de escalonamento e o simulamos.
- Como podemos ver no nosso PDF [link para tabela](#)

# Explicar código em Java

explicar