

# Sistemas Operacionais

Aula 10 – DIN00023

Capítulo 8 (MACHADO; MAIA, 2014)

Gerência do Processador

Prof. Dr. Jonathan Ramos  
[jonathan@unir.br](mailto:jonathan@unir.br)

**Departamento Acadêmico de Ciências de Computação – DACC**  
**Núcleo de Tecnologia – NT**

06/12/2022

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

# Introdução

## Sistemas

### multi-programáveis:

Diversos processos **concorrem** ao uso da UCP.

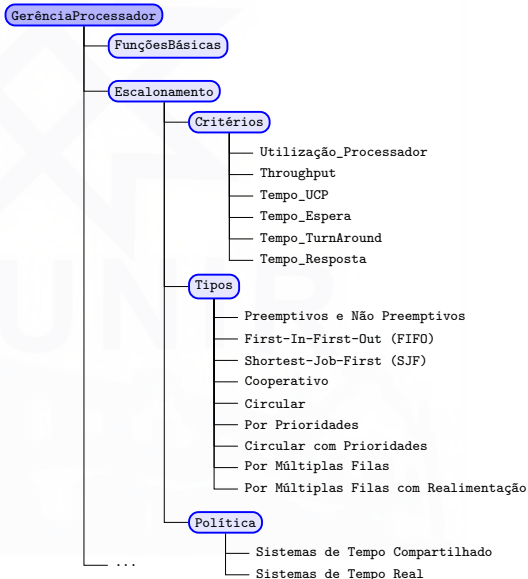
## Gerência do Processador:

Tornou-se **uma das atividades mais importantes**.

## Processos prontos:

Deve haver **critérios para execução**: política de escalonamento.

**Veremos os critérios e tipos de escalonamento nesta aula.**



# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

## Funções Básicas



## Funções Básicas

Uma das principais funções do SO é implementar os critérios da política de escalonamento:

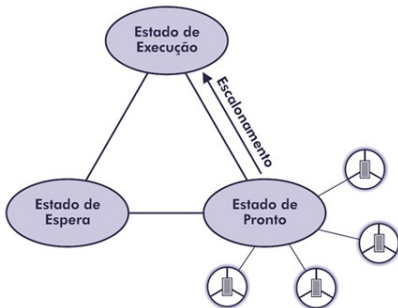


Figura: Escalonamento.

Em um sistema multiprogramável:

O escalonador (*scheduler*) é crucial: o compartilhamento da UCP depende desta rotina.

Dispatcher:

Responsável pela troca de contexto dos processos:

- Latência do *dispatcher*: O período de tempo que leva na troca de um processo em execução por outro.

Tipos de escalonamento:

- Apenas Processos: realizado com base nos processos prontos para execução.
- Com *Threads*: consideram *threads* no estado de pronto.

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios



## Critérios de Escalonamento

Depende de cada tipo de sistema, em geral segue os critérios:

- 1 **Utilização do processador:** **deve ficar ocupado a maior parte do tempo:**
  - Utilização na faixa dos 30% indica subutilização da UCP.
  - Utilização na faixa de 90% indica sobrecarga no sistema.
- 2 **Throughput:** **número de processos executados em um intervalo de tempo:**
  - Quanto maior, mais tarefas são executadas.
- 3 **Tempo de Processador / Tempo de UCP:** tempo que um processo leva **no estado de execução** durante seu processamento;
- 4 **Tempo de Espera:** tempo total que um processo **permanece na fila de pronto** durante seu processamento;
- 5 **Tempo de *Turnaround*:** tempo que um processo leva **da criação até o término;**
- 6 **Tempo de Resposta:** tempo decorrido **entre uma requisição ao sistema ou à aplicação e o instante em que a resposta é exibida**

### Em geral:

O escalonador busca **otimizar a utilização do processador**, aumentar o *throughput* e diminuir *turnaround*, tempo de espera e de resposta.

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento**
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

## Tipos de Escalonamento



## Não Preemptivos e Preemptivos

Indica se o processo pode ou não ser interrompido:

### Não Preemptivos

Quando um processo está em execução **nenhum evento externo** pode ocasionar a perda do uso do processador.

### Preemptivos

**O SO pode interromper um processo em execução** e passá-lo para o estado de pronto.

**Atualmente, a maioria dos sistemas operacionais implementa políticas de escalonamento preemptivas**

## First-In-First-Out (FIFO)



Figura: Escalonamento FIFO: **não preemptivo**.

## First-In-First-Out (FIFO)



Figura: Escalonamento FIFO: **não preemptivo**.

### Desvantagem:

- 1 não preemptivo;
- 2 Não melhora o tempo médio de espera dos processos;
- 3 Processos *CPU-bound* levam vantagem no uso do processador sobre processos *I/O-bound*

## Shortest-Job-First (SJF)

Seleciona o processo que tiver o menor tempo de UCP para executar:

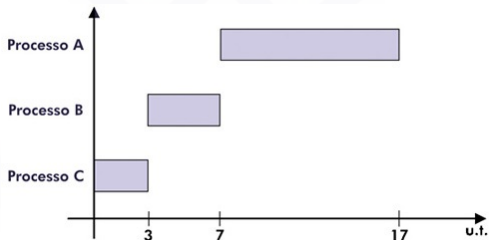


Figura: Escalonamento SJF (exemplo): **não preemptivo**.

### Desvantagem:

- 1 não preemptivo;
- 2 Processos longos podem sofrer *starvation*
- 3 O que mais?

## Cooperativo

### Aumentar o grau de multiprogramação em escalonadores não preemptivos:

#### Um processo em execução:

Pode voluntariamente liberar o processador.

- o processo em execução verifica periodicamente uma fila de mensagens para determinar se existem outros processos na fila de pronto

#### Algumas situações indesejadas podem acontecer:

- SO não gerencia a verificação: fica a cargo do processo em execução;
- **E se o processo em execução não fazer a verificação?**



## Circular

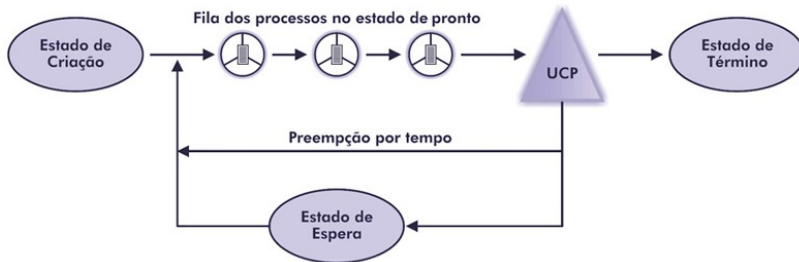


Figura: Escalonamento circular: **preemptivo**. Projetado especialmente para sistemas de tempo compartilhado

Similar ao FIFO:

Porém, quando um processo passa para o estado de execução existe **um tempo-limite para o uso contínuo do processador**:

- fatia de tempo (*time-slice*) ou *quantum*;
- preempção por tempo (**varia entre 10 e 100 ms**).
- **Vantagem? Desvantagem?**

## Circular

### Desvantagem:

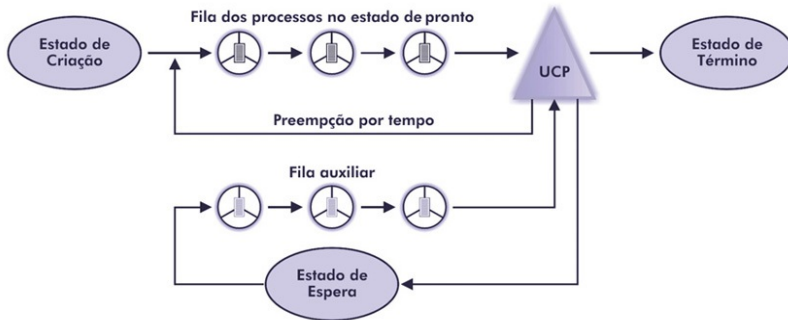
- Caso a fatia de tempo tenha um valor muito alto, este escalonamento **tenderá a ter o mesmo comportamento do escalonamento FIFO**;
- Caso o valor do *time-slice* seja pequeno, **a tendência é que haja um grande número de preempções**, o que ocasionaria excessivas mudanças de contexto, **prejudicando o desempenho do sistema e afetando o tempo de *turnaround* dos processos**.
- Processos *CPU-bound* **são beneficiados no uso do processador** em relação aos processos *I/O-bound*;
- Os processos *CPU-bound* **tendem a utilizar por completo a fatia de tempo**, enquanto os processos *I/O-bound* **têm mais chances de passar para o estado de espera antes de sofrerem preempção por tempo**;

### Vantagem:

- Não permitir que um processo monopolize a UCP;
- o escalonamento circular **é adequado para sistemas de tempo compartilhado**.

## Circular: refinamento

Uma lista extra mantém processos que saem do estado de espera e vão para uma fila de pronto:

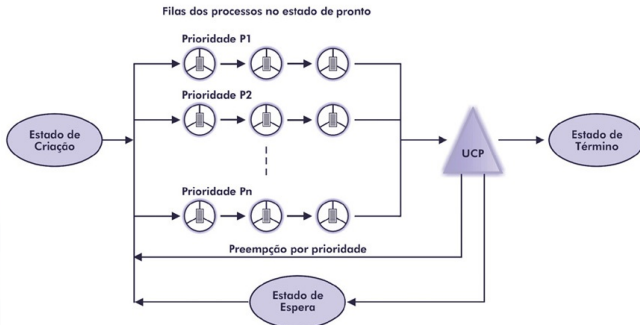


**Figura:** Escalonamento *circular virtual*: refinamento do escalonamento circular. Os processos da fila auxiliar **possuem preferência no escalonamento** em relação à fila de pronto. o escalonador **só seleciona processos na fila de pronto** quando a fila auxiliar estiver vazia

**Apesar da maior complexidade na implementação, esta abordagem apresenta resultados mais equilibrados: mecanismo adaptativo.**

## Por Prioridades

Cada processo possui um valor que representa sua prioridade de execução:



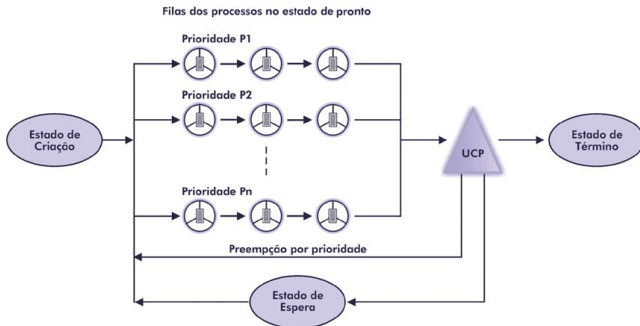
**Figura:** Escalonamento por prioridades: preemptivo. Processos com valores iguais são escalonados seguindo o critério de FIFO. O conceito de fatia de tempo não existe.

A perda do uso do processador **só ocorrerá no caso de uma mudança voluntária para o estado de espera** ou quando um processo de prioridade maior passa para o estado de pronto: **preempção por prioridade: implementada com interrupção de clock.**

**Desvantagem?**

## Por Prioridades

Cada processo possui um valor que representa sua prioridade de execução:



**Figura:** Escalonamento por prioridades: preemptivo. Processos com valores iguais são escalonados seguindo o critério de FIFO. O conceito de fatia de tempo não existe.

A perda do uso do processador **só ocorrerá no caso de uma mudança voluntária para o estado de espera** ou quando um processo de prioridade maior passa para o estado de pronto: **preempção por prioridade: implementada com interrupção de clock.**

**Desvantagem? Starvation!**

## Circular com Prioridades

**Adiciona fatias de tempo a cada processo:**

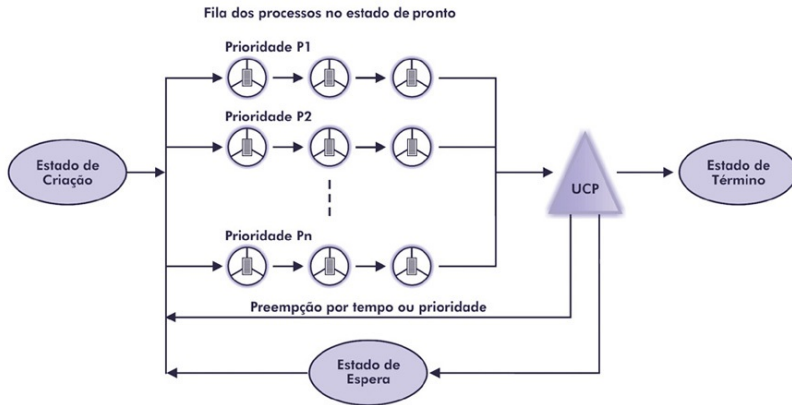


Figura: Escalonamento circular com prioridades.

**Atualmente é amplamente utilizado no sistemas Unix (Linux) e Windows.**

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

## Sistemas de Tempo Compartilhado

### Caracterizam-se pelo processamento interativo

Usuários interagem com as aplicações exigindo **tempos de respostas baixos**:

- Leva em consideração o compartilhamento dos recursos de **forma equitativa** para possibilitar o **uso balanceado da UCP** entre processos.

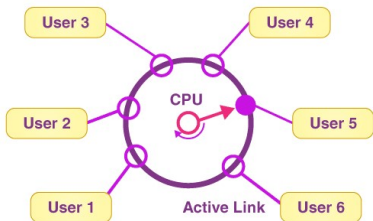


Figura: Multi-task ou time-sharing.

### Vantagens:

- Resposta rápida, CPU menos ociosa

### Desvantagem:

- Usa muitos recursos, necessita de hardware mais potente, maiores preocupações com segurança e comunicação.

**Atualmente, a maioria dos SOs de tempo compartilhado utiliza o escalonamento circular com prioridades dinâmicas**



## Sistemas de Tempo Real

**Algumas aplicações específicas exigem respostas imediatas para a execução de determinadas tarefas**

SO de tempo real:

É garantida a execução de processos **dentro de limites rígidos de tempo**, sem o risco de a aplicação ficar comprometida.

**Exemplos?**

## Sistemas de Tempo Real

**Algumas aplicações específicas exigem respostas imediatas para a execução de determinadas tarefas**

SO de tempo real:

É garantida a execução de processos **dentro de limites rígidos de tempo**, sem o risco de a aplicação ficar comprometida.

### Exemplos?

- Controle de tráfego aéreo, Robótica, Visão Computacional, etc

### Escalonamento:

Deve levar em consideração **a importância relativa de cada tarefa na aplicação**:

- O escalonamento por prioridades **é o mais adequado**;
- Cada processo tem uma prioridade que **é associada em função da importância do processo** dentro da aplicação;
- **Não deve existir o conceito de fatia de tempo**, e a prioridade de cada processo deve ser estática
- em **algumas sub-faixas de prioridade** não há preempção por tempo.

# Sumário

- 1 Introdução
- 2 Funções Básicas
- 3 Critérios de Escalonamento
- 4 Tipos de Escalonamento
  - Não Preemptivos e Preemptivos
  - First-In-First-Out (FIFO)
  - Shortest-Job-First (SJF)
  - Cooperativo
  - Circular
  - Por Prioridades
  - Circular com Prioridades
- 5 Política de Escalonamento
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- 6 Exercícios

## Exercícios I

- 1 O que é política de escalonamento de um sistema operacional?
- 2 Quais as funções do escalonador e do *dispatcher*?
- 3 Quais os principais critérios utilizados em uma política de escalonamento?
- 4 Diferencie os tempos de processador, espera, *turnaround* e resposta.
- 5 Diferencie os escalonamentos preemptivos e não preemptivos.
- 6 Qual a diferença entre os escalonamentos FIFO e circular?
- 7 Descreva o escalonamento SJF e o escalonamento por prioridades.
- 8 Qual a diferença entre preempção por tempo e preempção por prioridade?
- 9 O que é um mecanismo de escalonamento adaptativo?
- 10 Que tipo de escalonamento aplicações de tempo real exigem? Justifique.

## Exercícios II

- 11 Considere que cinco processos sejam criados no instante de tempo 0 (P1, P2, P3, P4 e P5) e possuam as características descritas na tabela a seguir:

Processo	Tempo de UCP	Prioridade
P1	10	3
P2	14	4
P3	5	1
P4	7	2
P5	20	5

Desenhe um diagrama ilustrando o escalonamento dos processos e seus respectivos tempos de *turnaround*, segundo as políticas especificadas a seguir. O tempo de troca de contexto deve ser desconsiderado.

- 1 FIFO
- 2 SJF
- 3 Prioridade (número menor implica prioridade maior)
- 4 Circular com fatia de tempo igual a 2 u.t.

## Exercícios III

- 12 Considere um sistema operacional com escalonamento por prioridades onde a avaliação do escalonamento é realizada em um intervalo mínimo de 5 ms. Neste sistema, os processos A e B competem por uma única UCP. Desprezando os tempos de processamento relativo às funções do sistema operacional, a tabela a seguir fornece os estados dos processos A e B ao longo do tempo, medido em intervalos de 5 ms (E = execução, P = pronto e W = espera). O processo A tem menor prioridade que o processo B.

	00-04	05-09	10-14	15-19	20-24	25-29	30-34	35-39	40-44	45-49
<b>Processo A</b>	P	P	E	E	E	P	P	P	E	W
<b>Processo B</b>	E	E	W	W	P	E	E	E	W	W

	50-54	55-59	60-64	65-69	70-74	75-79	80-84	85-89	90-94	95-99	100-105
<b>Processo A</b>	P	E	P	P	E	E	W	W	P	E	E
<b>Processo B</b>	W	P	E	E	W	W	P	E	E	–	–

- 1 Em que tempos A sofre preempção?;
- 2 Em que tempos B sofre preempção?
- 3 Refaça a tabela anterior supondo que o processo A é mais prioritário que o processo B.

# Laboratório com o Simulador SOsim: Atividade 1: Escalonamento Circular

- 1 **Práticas de simulação:** Execute o simulador SOsim e configure-o para trabalhar com Escalonamento Circular: janela Console SOsim/Opções/Parâmetros do Sistema na guia Processador.
- 2 **Análise prática:**
  - Crie dois processos com a mesma prioridade (um *CPU-bound* e outro *I/O-bound*): janela Gerência de Processos / Criar – janela Criação de Processos / Criar.
  - Na janela Gerência de Processos, observe o tempo de processador de cada processo durante dois minutos e as mudanças de estado. Após esse período, anote o tempo de processador de cada processo. Analise o balanceamento no uso do processador pelos dois processos.
  - Na janela Gerência de Processos, finalize os dois processos.
  - Na janela Gerência de Processador, aumente a fatia de tempo movimentando a barra de Fatia de Tempo.
  - Na janela Gerência de Processos, observe mais uma vez o tempo de processador de cada processo durante dois minutos e as mudanças de estado. Após esse período, anote o tempo de processador de cada processo. Compare os tempos anotados nas duas e analise o resultado do balanceamento no uso do processador pelos dois processos. Identifique as causas da variação.
- 3 **Questão teórica para responder com a ajuda do simulador:** Considere a concorrência, nesse tipo de escalonamento, com dois processos *CPU-bound* que não realizam operações de E/S. Qual o efeito da variação da fatia de tempo sobre o balanceamento no uso do processador?

## Laboratório com o Simulador SOsim: Atividade 2: Escalonamento Circular com Prioridades Estáticas I

- 1 **Práticas de simulação:** Execute o simulador SOsim e configure-o para trabalhar com Escalonamento Circular com Prioridades Estáticas: janela Console SOsim/Opções/Parâmetros do Sistema na guia Processador.
- 2 **Análise prática:**
  - Crie um processo *CPU-bound* com prioridade 3 e um outro *I/O-bound* com prioridade 4: janela Gerência de Processos/Criar – janela Criação de Processos/Criar.
  - Na janela Gerência de Processos, observe o tempo de processador de cada processo durante dois minutos e as mudanças de estado. Após esse período anote o tempo de processador de cada processo.
  - Verifique a preempção por prioridade que ocorre toda vez que o processo *I/O-bound* de maior prioridade passa para o estado de pronto.
  - Analise o balanceamento no uso do processador pelos dois processos comparativamente à Atividade 1.
- 3 **Questão teórica para responder com a ajuda do simulador:** Quais devem ser os critérios para determinar as prioridades dos processos? Caso, nesse escalonamento, todos os processos sejam criados com a mesma prioridade, qual o benefício dessa política sobre o Escalonamento Circular?



## Laboratório com o Simulador SOsim: Atividade 3: Escalonamento Circular com Prioridades Estáticas II

- 1 **Práticas de simulação:** Execute o simulador SOsim e configure-o para trabalhar com Escalonamento Circular com Prioridades Estáticas: janela Console SOsim/Opções/Parâmetros do Sistema na guia Processador.
- 2 **Análise prática:**
  - Crie um processo *CPU-bound* com prioridade 4 e um outro *I/O-bound* com prioridade 3: janela Gerência de Processos/Criar – janela Criação de Processos/Criar.
  - Na janela Gerência de Processos, observe o escalonamento dos dois processos. Analise o problema do *starvation*.
- 3 **Questão teórica para responder com a ajuda do simulador:** Por que o problema do *starvation* pode ocorrer? Cite duas ações que o administrador do sistema pode realizar quando é identificada a situação de *starvation* em um processo.

# Laboratório com o Simulador SOsim: Atividade 4: Escalonamento Circular com Prioridades Dinâmica

- 1 **Práticas de simulação:** Execute o simulador SOsim e configure-o para trabalhar com Escalonamento Circular com Prioridades Dinâmicas: janela Console SOsim/Opções/Parâmetros do Sistema na guia Processador. Habilite as janelas de log e estatísticas: janela Console SOsim/Janelas. Na janela Gerência do Processador desloque a barra Frequência clock para a metade da escala.
- 2 **Análise prática:**
  - Crie um processo CPU-bound com prioridade base 3 e mais três processos I/O-bound com prioridade base 4, porém com perfis diferentes (tipo 1, 2 e 3): janela Gerência de Processos/Criar — janela Criação de Processos/Criar.
  - Observe as prioridades base e dinâmica dos quatro processos na janela Gerência de Processos.
  - Identifique os motivos de as prioridades dinâmicas dos processos variarem ao longo do tempo.
  - Observe na janela de log o valor do incremento recebido na prioridade de cada processo,
  - Identifique o porquê das diferenças nos valores do incremento.
  - Observe na janela de estatísticas o percentual de utilização da UCP.
  - Suspenda o processo CPU-bound: janela Gerência de Processos/Suspender.
  - Observe na janela de estatísticas as mudanças no percentual de utilização da UCP e identifique a razão.
  - Libere o processo CPU-bound do estado de suspensão: janela Gerência de Processos/Prosseguir.
- 3 **Questão teórica para responder com a ajuda do simulador:** Qual o critério utilizado pelo sistema operacional para determinar diferentes valores de incremento à prioridade-base de um processo quando há uma mudança do estado de espera para pronto?

## Referências I



MACHADO, F. B.; MAIA, L. P. **Arquitetura de Sistemas Operacionais**.  
5a. ed. [S.l.]: Ed. LTC, 2014.

**FIM!**

`jonathan@unir.br`