

# Sistemas Operacionais

## Gerenciamento de Processos

*Profa: Deborah Magalhães*



“

PROCESSO:

Dá suporte à realização de operações (**pseudo**) CONCORRENTES mesmo quando há apenas uma CPU disponível, transformando uma **única** CPU em **múltiplas** CPUs virtuais.

**CENÁRIO:**  
Solicitações de páginas  
Web chegam de toda parte  
ao servidor

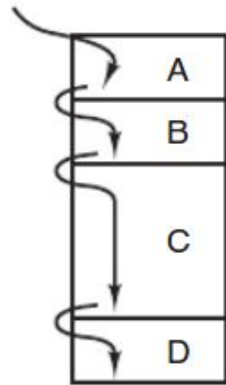
“

## MODELO DE PROCESSOS:

É um modelo conceitual, onde **todos** os softwares executados no computador, incluindo o sistema operacional, são organizados em uma série de processos SEQUENCIAIS

# Multiprogramação

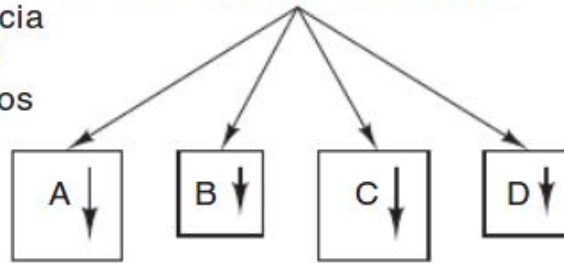
Um contador de programa



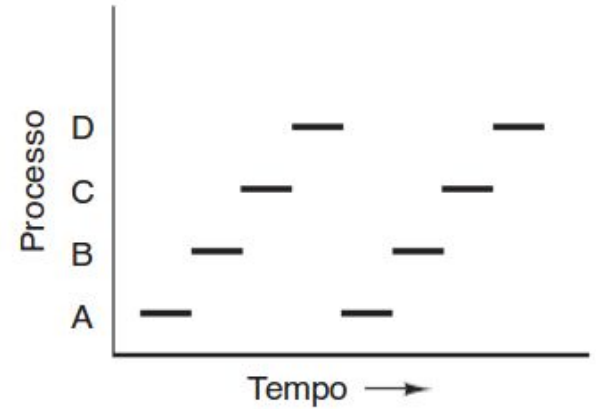
(a)

Alternância  
entre  
processos

Quatro contadores de programa



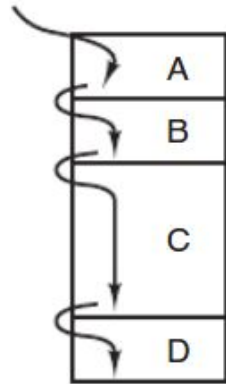
(b)



(c)

# Multiprogramação

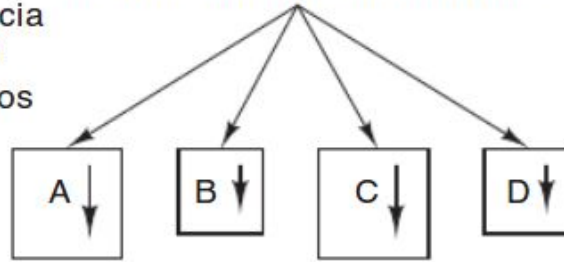
Um contador de programa



(a)

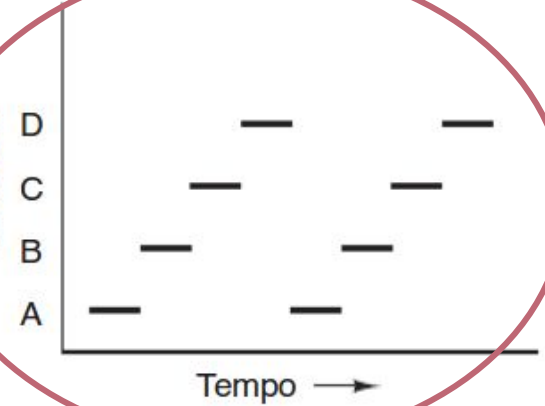
Alternância  
entre  
processos

Quatro contadores de programa



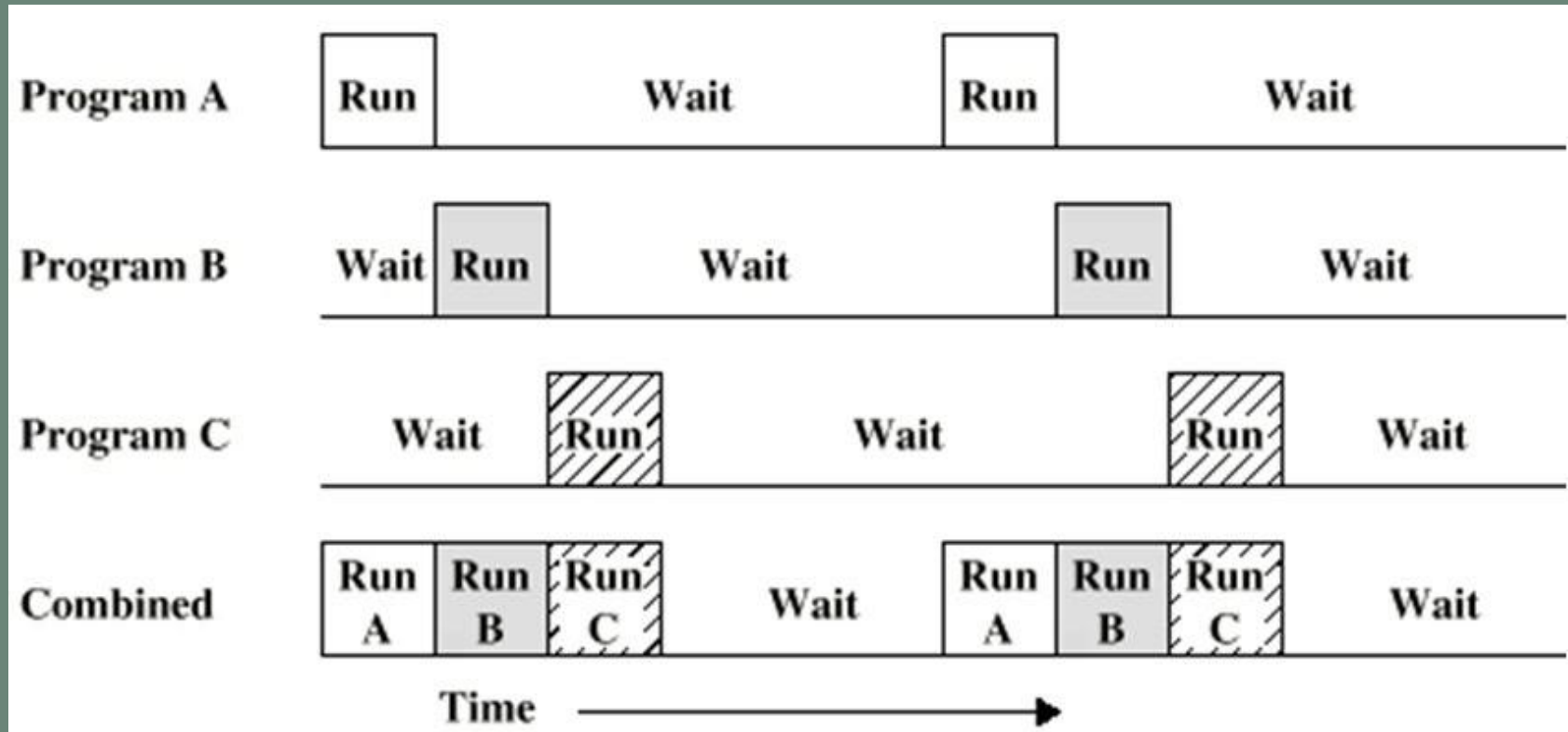
(b)

Processo

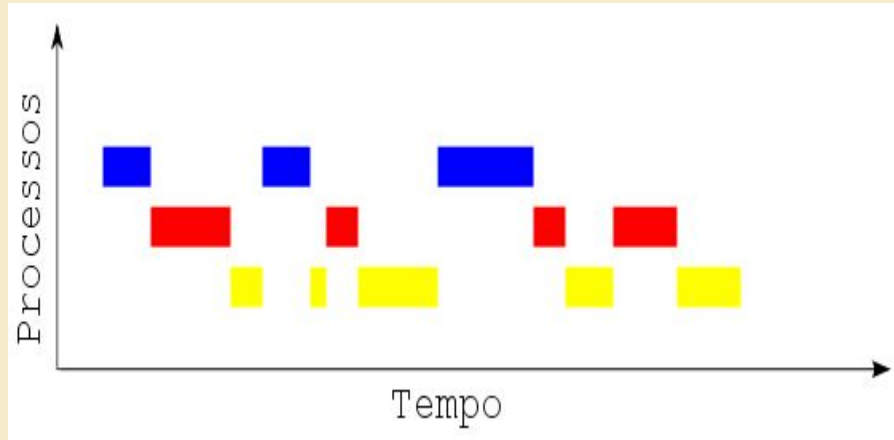


(c)

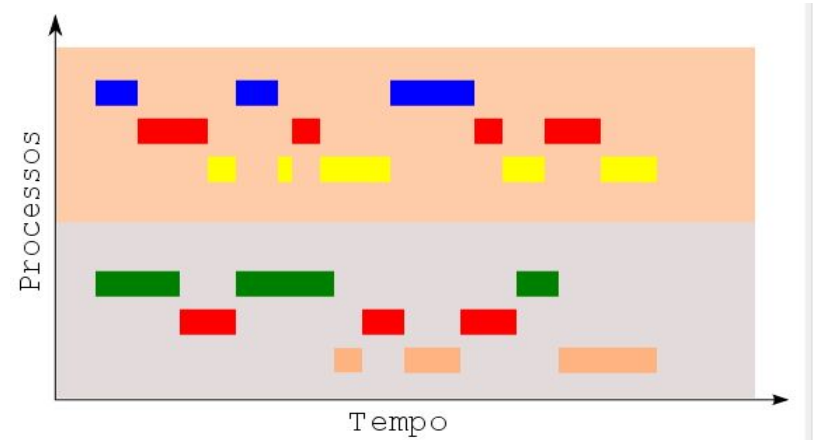
# Multiprogramação



## Concorrência



## Paralelismo





# **CRIAÇÃO** de processos

## **Inicialização do sistema**

GUI (front) e daemons (background)

## **Chamada de sistema**

Fork e CreateProcess

## **Solicitação de usuário**

GUI ou linha de comando

## **Início de uma tarefa em lote**

# **TÉRMINO** de processos

## **Saída voluntária**

GUI ou chamada de sistema (exit, ExitProcess)

## **Erro Fatal (involuntário)**

## **Saída por erro (voluntária)**

Execução de uma instrução ilegal

## **Morto por processo (involuntário)**

Chamada de sistema (kill, TerminateProcess )



# Hierarquia de Processos

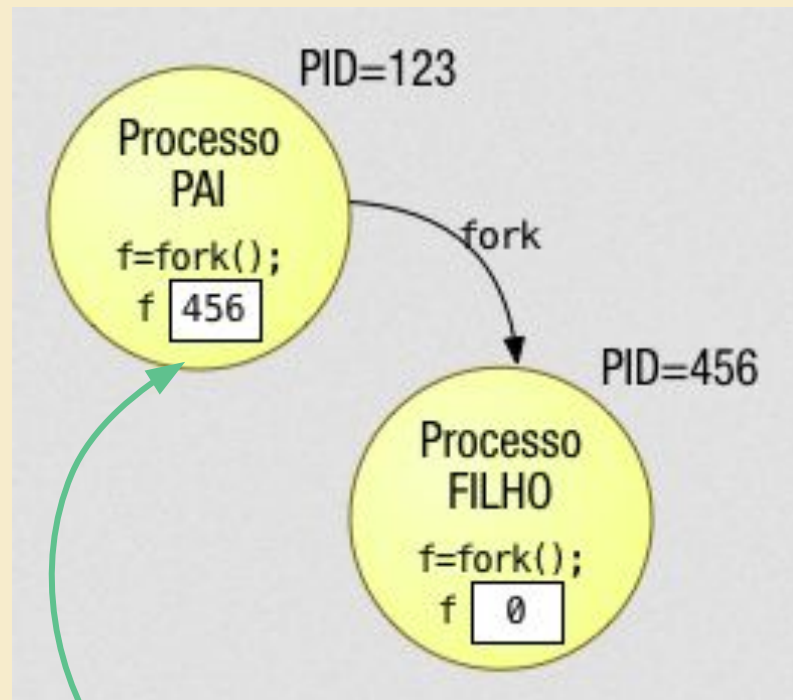
*Cada processo possui um único pai, exceto o nó raiz*



# Hierarquia de Processos

Pai e filho **NÃO** compartilham:

- PID
- Espaço de endereçamento
  - Constantes e Variáveis
  - Pilha de execução



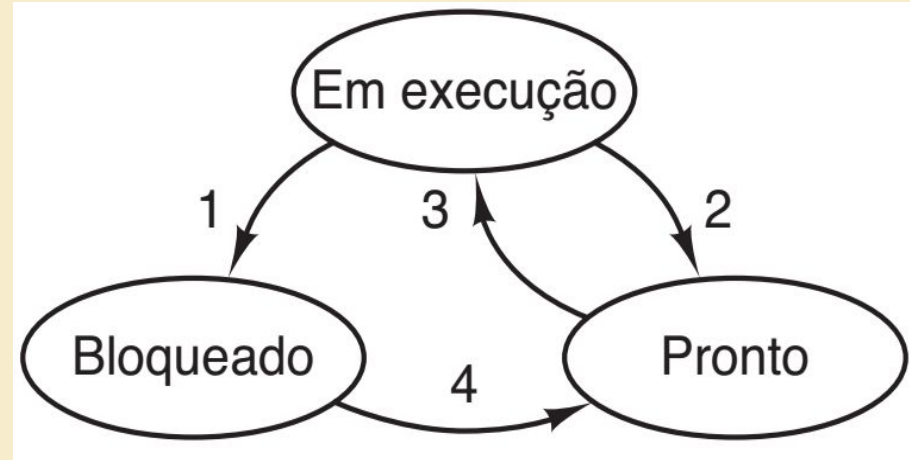
*Referência ao filho*

## QUESTÃO:

Windows e Linux possuem o mesmo comportamento em relação a hierarquia de processos?

# Estados do Processo

1. O processo é bloqueado aguardando uma entrada
2. O escalonador seleciona outro processo
3. O escalonador seleciona esse processo
4. A entrada torna-se disponível



## Bloqueado

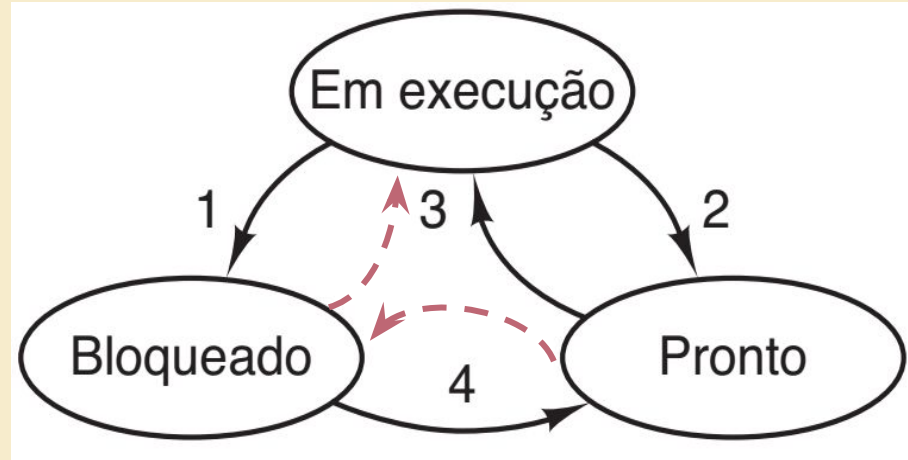
```
$ cat cap1 cap2 cap3 | grep processo
```

## Pronto

O processo pode executar a qualquer instante MAS não pode acessar a **CPU** porque **OUTRO** processo está utilizando

# Estados do Processo

1. O processo é bloqueado aguardando uma entrada
2. O escalonador seleciona outro processo
3. O escalonador seleciona esse processo
4. A entrada torna-se disponível

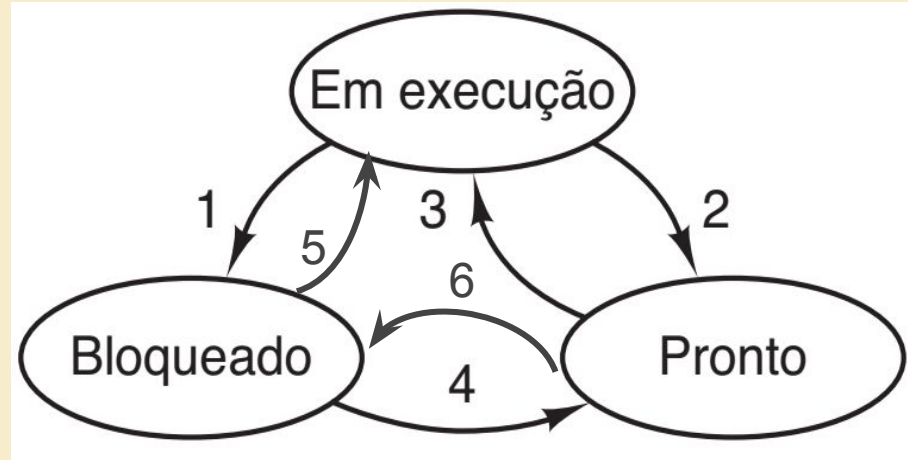


*Por que as linhas tracejadas não fazem parte do ciclo de vida do processo?*



# Estados do Processo

1. O processo é bloqueado aguardando uma entrada
2. O escalonador seleciona outro processo
3. O escalonador seleciona esse processo
4. A entrada torna-se disponível

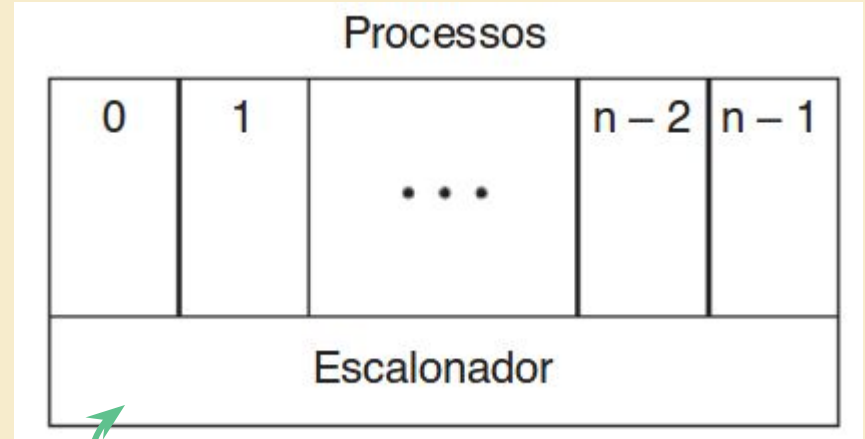


*Por que as linhas tracejadas não fazem parte do ciclo de vida do processo?*

# Escalonador

Responsável por tratar:

- Interrupções
- Início e fim de processo
- Chaveamento de processos na CPU



*Nível mais baixo do SO*



# Muito Obrigada!

Se você tiver qualquer dúvida ou sugestão:

- [deborah.vm@ufpi.edu.br](mailto:deborah.vm@ufpi.edu.br)

