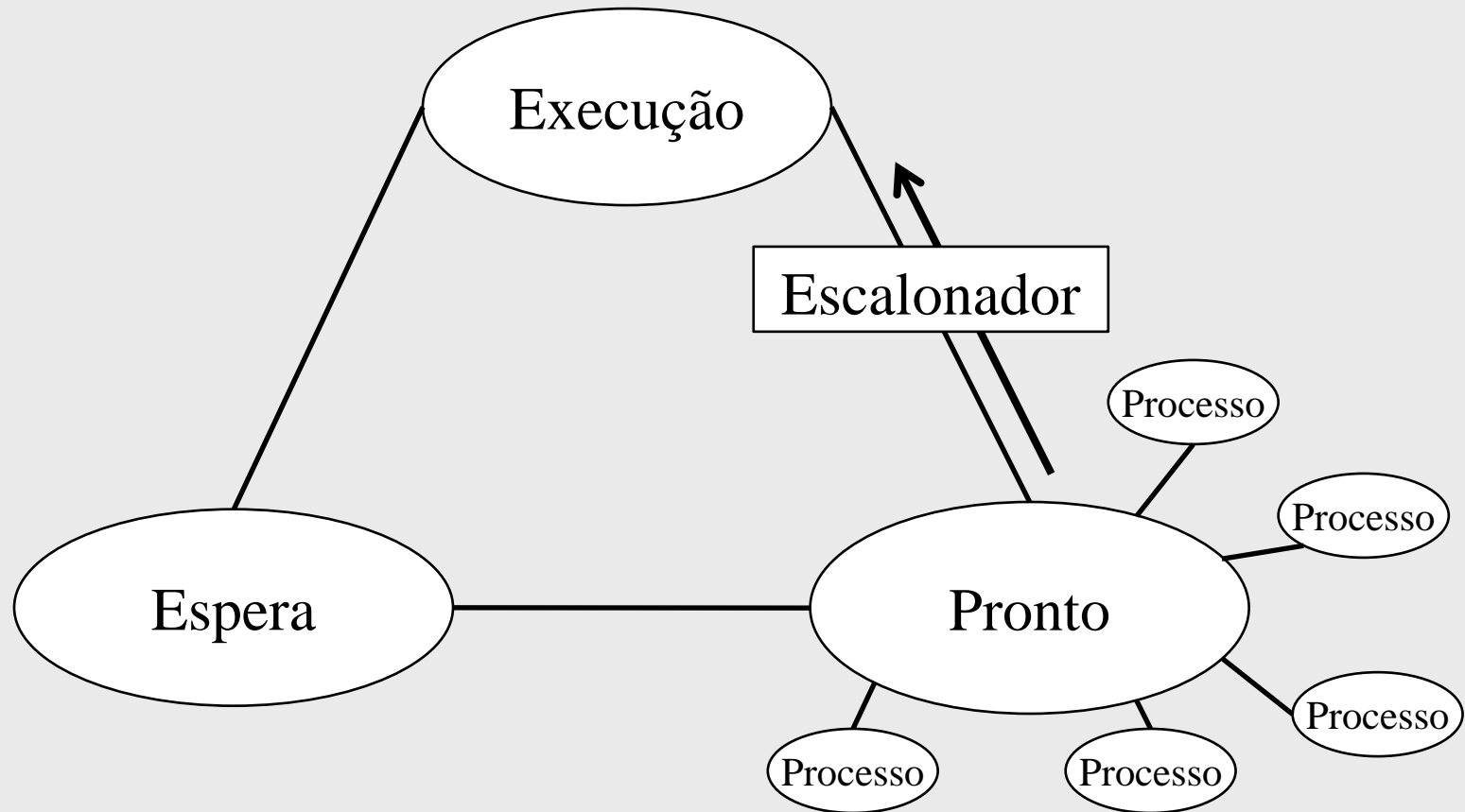


Danton Cavalcanti Franco Junior
falecom@dantonjr.com.br

Escalonamento de Processos

- É a tarefa de decidir qual o processo irá ocupar o processador quando este ficar livre, ou seja é a parte do S.O. encarregada de tomar as decisões de escolha.
- Escalonamento – Scheduling.

Escalonamento de Processos



Escalonamento de Processos

□ Objetivos:

- Manter a CPU ocupada.
- Balancear a utilização do processador.
- Maximizar o throughput e oferecer tempos de resposta razoáveis para usuários interativos.
- Tratar todos os processos igualmente (evitar o starvation).

Escalonamento de Processos

□ Objetivos:

- Menor Overhead: Tempo de gerência (troca dos processos).
- Equilíbrio entre resposta e utilização.
- Garantia de prioridades.
- Tempo de turnaround: Determina o tempo desde a admissão até o término do processo.

Escalonamento de Processos

- ❑ O Escalonador decide baseado em uma política de escolha utilizando os algoritmos de escalonamento.
- ❑ Problemas com o comportamento de cada um dos processos. Muita I/O ou muita utilização do processador.

Tipos

□ **Escalonamento Não-Preemptivo:**

Estratégia de rodar o processo até o fim (batch), ou seja após o processamento ganhar a CPU ninguém pode interrompê-lo.

Escalonamento Preemptivo: Estratégia de permitir a suspensão temporária dos processos.

Não-Preemptivo

- ❑ **Escalonamento First-In-First-Out – FIFO**
- ❑ O processo que chegar primeiro é o primeiro a ser selecionado.
- ❑ Algoritmo simples (possui apenas uma lista).
- ❑ Processos em estado de pronto passam para o fim da lista e são escalonados posteriormente.
- ❑ Uma vez que o processo está com a CPU ele é executado até a sua conclusão.
- ❑ Não se pode prever quando o processo será executado (depende dos que estão na frente).

Não-Preemptivo

- ❑ **Escalonamento Shortest-Job-First – SJF**
- ❑ Tarefas com menor tempo de utilização são executadas por primeiro.
- ❑ Objetivo de livrar-se do maior número de processos o quanto antes.
- ❑ Dificuldade em calcular quanto tempo a tarefa irá utilizar a CPU (produção é mais fácil determinar que em ambientes de desenvolvimento).

Não-Preemptivo

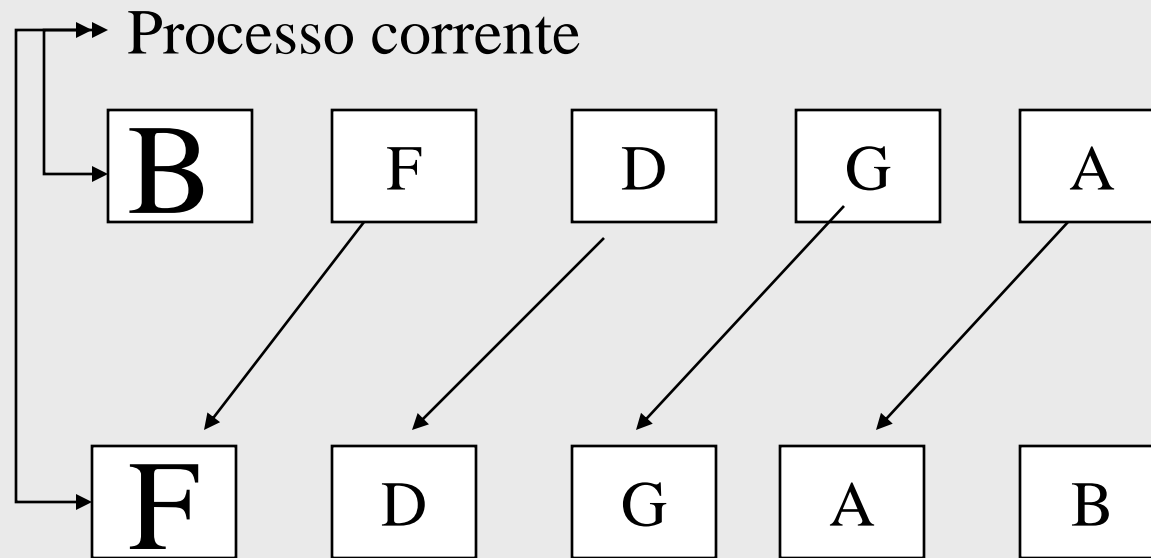
- ❑ **Escalonamento Cooperativo**
- ❑ Processo em execução entrega o processador voluntariamente sem a intervenção do S.O.
- ❑ Processos podem entrar em looping e monopolizar a CPU.
- ❑ Usado no Windows 3.1.
- ❑ Processos ficam “escutando” a fila de mensagens para ver solicitações de outros processos (se não ouvir, vai até o fim).

Preemptivo

- ❑ **Escalonamento Circular (Round Robin)**
- ❑ A cada processo atribui-se um intervalo de tempo durante o qual ele poderá utilizar o processador (**quantum** ou **time-slice**).
- ❑ Se esgotar o seu quantum ele perde o processador (volta para o fim da fila).
- ❑ Manter uma lista de processos prontos.
- ❑ Nenhum processo pode monopolizar a CPU.

Preemptivo

□ Round Robin



Preemptivo

- **Round Robin**
- Determinação do tamanho do quantum.
 - **Muito pequeno:** há sucessivas trocas de contexto baixando a eficiência do processador (overhead).
 - **Muito Grande:** pode levar a um tempo de resposta não aceitável.

Preemptivo

- ❑ **Escalonamento por Prioridades**
- ❑ Processos devem ser tratados de maneira diferente dos outros.
- ❑ Os processos de maior prioridades são escalonados antes.
- ❑ Sistemas de tempo compartilhado implementam algum tipo de prioridade de forma a dar mais importância na hora do escalonamento.

Preemptivo

- ❑ **Escalonamento por Prioridades**
- ❑ Processos I/O-bound devem ganhar mais tempo que CPU-bound (por esperarem muito).
- ❑ A prioridade pode ser:
 - Estática: não é modificada durante a existência do processo.
 - Dinâmica: pode ser alterada conforme a necessidade do sistema.

Preemptivo

- ❑ **Escalonamento por Prioridades**
- ❑ Todo processo ao sair do estado de espera recebe um acréscimo em sua prioridade. Dessa forma, os processos IO-bound terão mais chances de serem escalonados, e assim, compensar o tempo que passam esperando. Processos CPU-bound não são prejudicados, pois podem executar enquanto processos IO-bound esperam algum evento.

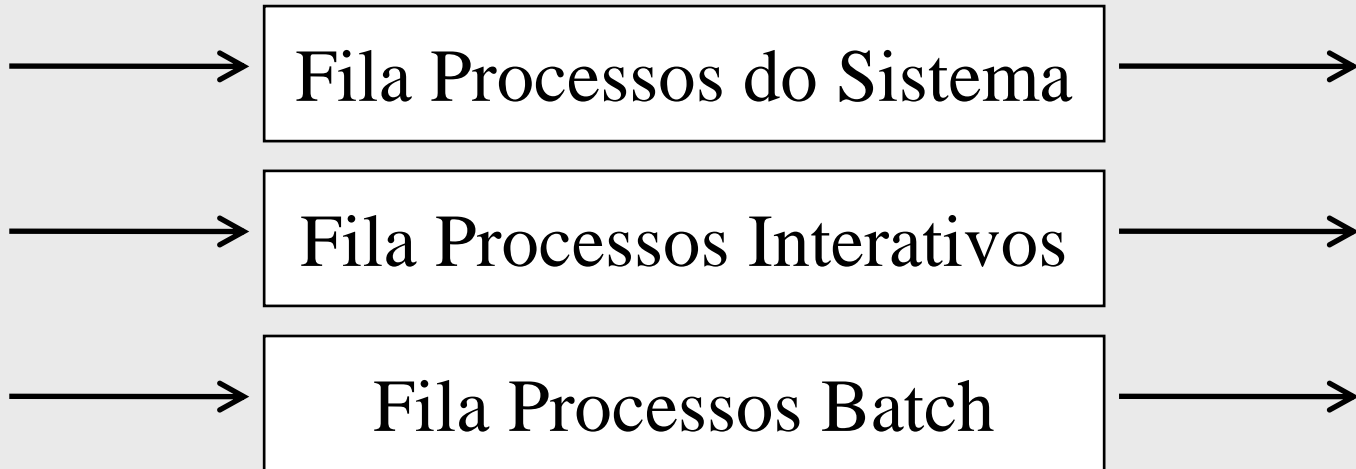
Preemptivo

- ❑ **Escalonamento por Múltiplas Filas**
- ❑ Multi-level queues.
- ❑ Os processos são classificados em função do tipo de processamento realizado e a cada grupo criado, são aplicados mecanismos de escalonamento distintos.
- ❑ Cada fila possui uma prioridade associada, que só executa quando as prioridades maiores estão vazias.

Preemptivo

□ Escalonamento por Múltiplas Filas

Maior prioridade



Menor prioridade

Preemptivo

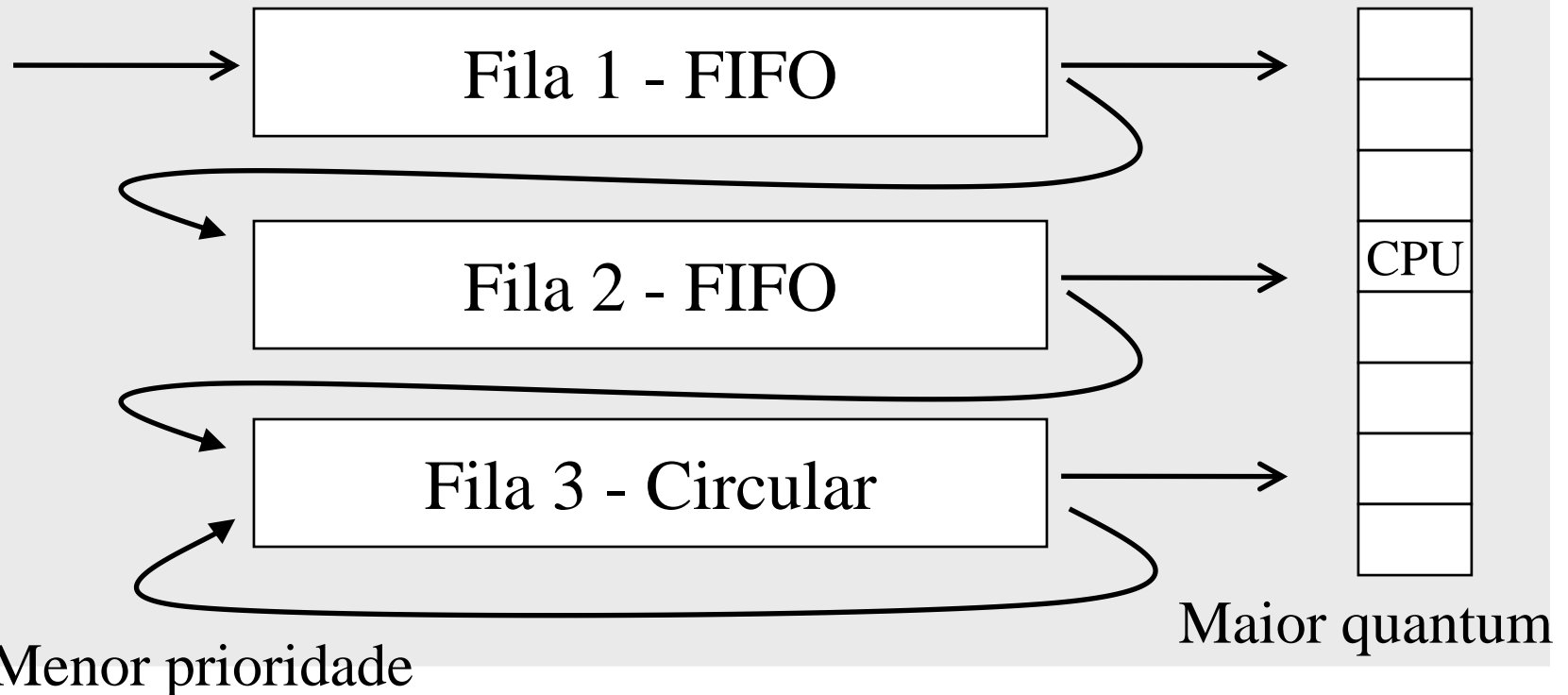
- **Escalonamento por Múltiplas Filas com Realimentação**
 - Multi-level feedback queues.
 - Possuem prioridade de execução mas não necessariamente permanecem até o final de seu processamento.
 - *Mecanismo adaptativo*, que tem como objetivo ajustar os processos em função do comportamento do sistema.

Preemptivo

□ Escalonamento por Múltiplas Filas com Realimentação

Maior prioridade

Menor quantum



Preemptivo

- **Escalonamento de Sistemas de Tempo Real**
 - O fator tempo é crítico.
 - Não existe o conceito de *quantum*.
 - Para cada processo é atribuída uma prioridade associada a sua importância dentro do sistema.
 - A prioridade é estática.
 - Quanto maior a importância de uma tarefa, maior sua prioridade de execução.

Preemptivo

- **Escalonamento com Múltiplos Processadores**
 - Em sistemas fracamente acoplados, cada processador faz seu próprio escalonamento local.
 - Nos sistemas fortemente acoplados é possível implementar uma única *fila de pronto* para todos os processadores.

Preemptivo

- **Escalonamento com Múltiplos Processadores**
 - Todos os processos estão presentes nesta única fila e são escalonados no primeiro processador disponível.
 - Deve ser implementada a exclusão mútua.
 - A exclusão mútua do Escalonador pode ser obtida através de mecanismos como semáforos e monitores.