

**Universidade Federal do Rio Grande do Norte
Unidade Acadêmica Especializada em Ciências Agrárias
Escola Agrícola de Jundiaí
Curso de Análise e Desenvolvimento de Sistemas
TAD0006 - Sistemas Operacionais - Turma 01**

Processos - Escalonamento

Antonino Feitosa
antonino.feitosa@ufrn.br

Macaíba, maio de 2025



Aula Passada

- Condições de Corrida
- Regiões Críticas
- Exclusão Mútua com Espera Ocupada
- Dormir e Acordar
- Semáforos

Roteiro

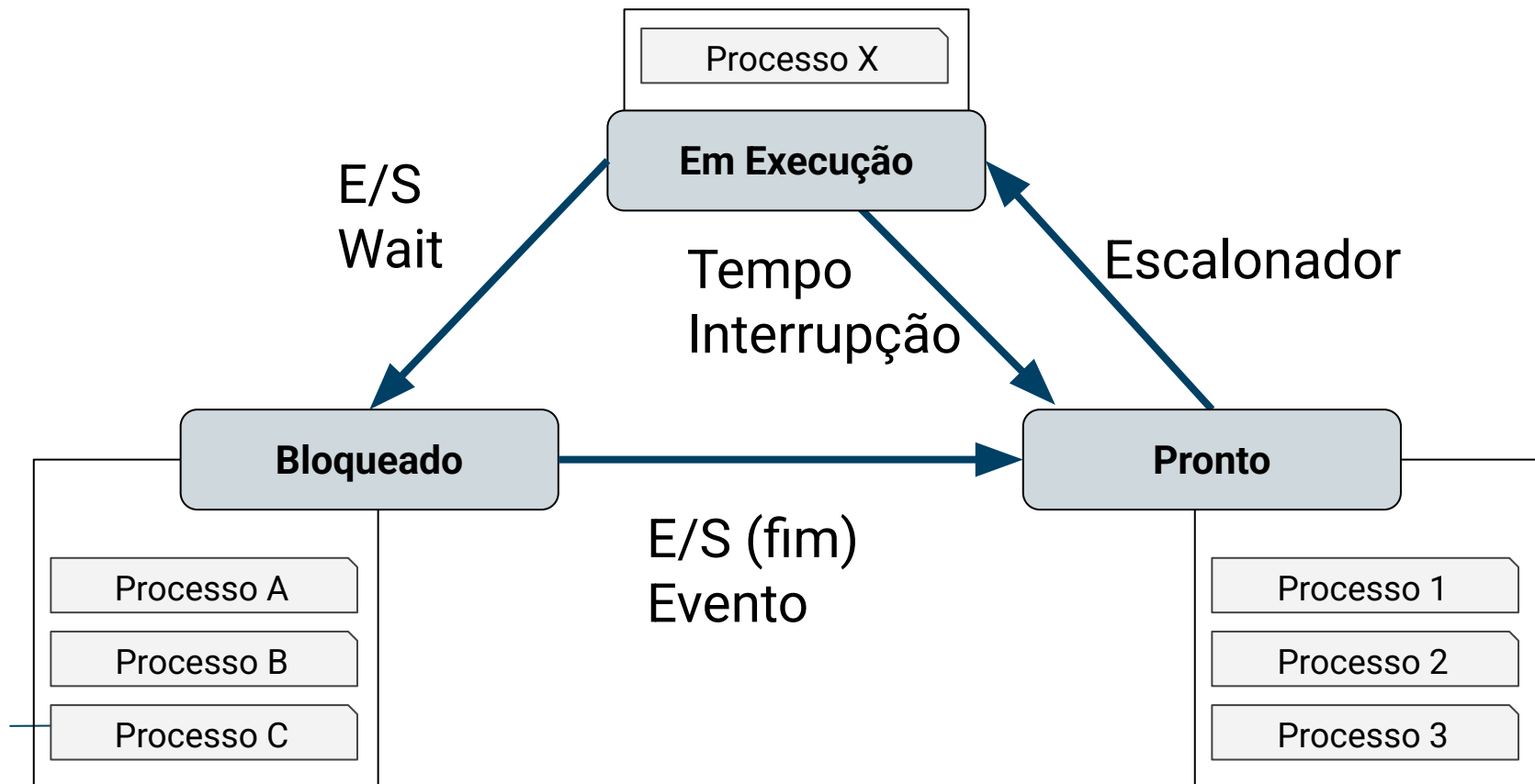
- Fundamentos
- Tipos e Objetivos dos Algoritmos
- Algoritmos de Escalonamento
 - Escalonamento em Sistemas em Lote
 - Escalonamento em Sistemas Interativos
- Política versus Mecanismos

Escalonamento

Escalonamento

- Consequência da multiprogramação: múltiplos processos ou threads competem pela CPU.
 - Processos no estado de pronto.
- Escalonador: parte do sistema operacional que escolhe qual processo no estado de pronto deve ser executado em seguida.
 - Critério: algoritmo do escalonador.

Escalonamento



Escalonamento

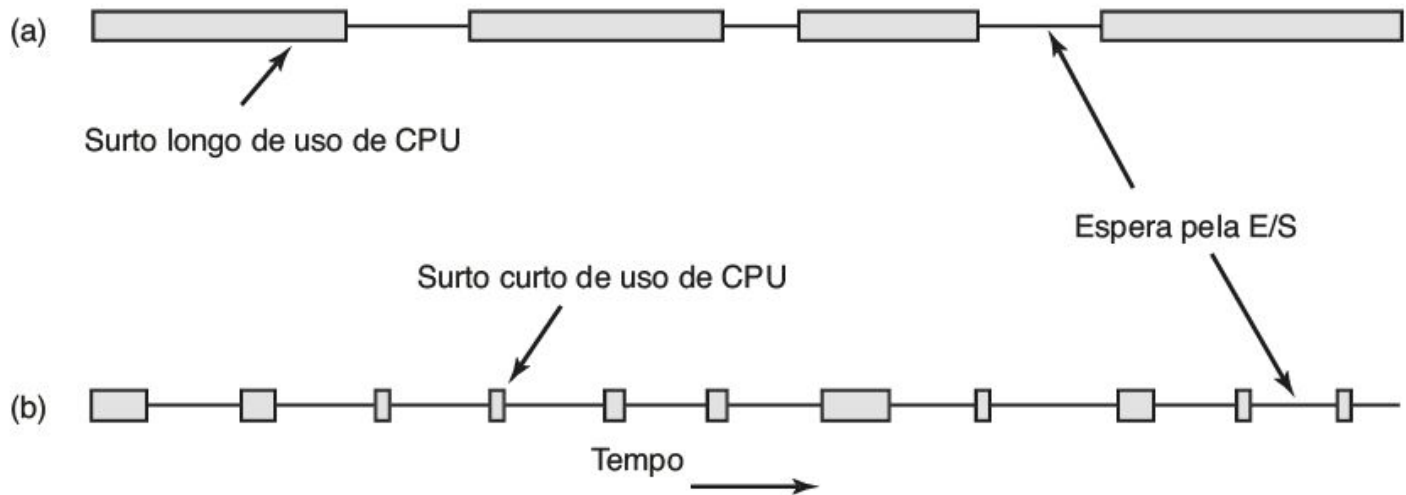
- Conceitos podem ser aplicados ao escalonamento de threads.
 - Quando a implementação das threads é a nível de núcleo, o escalonamento não considera o processo ao qual a thread pertence.

Escalonamento

- Comportamento dos processos.
 - CPU-bound (limitados pelo CPU ou pela computação): passam a maior parte do tempo efetuando computações.
 - Surtos longos de uso da CPU, com esperas ocasionais por E/S.
 - IO-bound (limitados pela E/S): passam a maior parte do tempo esperando por operações de E/S.
 - Surtos curtos de uso da CPU, com esperas frequentes por E/S.

Escalonamento

FIGURA 2.39 Surtos de uso da CPU alternam-se com períodos de espera por E/S. (a) Um processo limitado pela CPU. (b) Um processo limitado pela E/S.



Escalonamento

- Observação: à medida que as CPUs ficam mais rápidas, os processos tendem a ficar mais limitados pela E/S.
 - Escalonamento de processos limitados pela E/S torne-se um assunto mais importante no futuro.
 - Se um processo limitado pela E/S quiser executar, ele deve receber uma chance rapidamente para que possa emitir sua solicitação de disco e manter o disco ocupado.

Escalonamento: Quando Escalonar?

- Quando um processo é criado.
 - O processo pai e o filho estão em estado de pronto.
- Quando um processo termina.
 - Criado processo ocioso na ausência de processos no estado de pronto.
- Quando um processo é bloqueado.
 - Exemplo da região crítica.
- Quando ocorre uma interrupção de E/S indicando que um dispositivo completou uma tarefa.

Escalonamento: Quando Escalonar?

- Podemos considerar interrupções do relógio para guiar eventos de escalonamento — interrupções periódicas.
- Algoritmos de escalonamento:
 - **Preemptivo**: escolhe um processo e o deixa executar por no máximo um certo tempo fixado.
 - Necessita das interrupções de relógio!
 - **Não preemptivo**: escolhe um processo para ser executado e então o deixa ser executado até que ele seja bloqueado, ou libera voluntariamente a CPU.
 - Associado à prioridades.

Escalonamento: Consequências

- A troca de contexto (chaveamento de processos) é procedimento custoso em termos de processamento:
 - Deve ocorrer no modo de núcleo
 - Registro do estado do processo na memória.
 - A memória precisa ser atualizada com o contexto do novo processo.
 - Limpeza de cache e carregamento de dados da memória principal.
- Trocas sucessivas de processos em um intervalo curto de tempo pode causar uma sobrecarga no sistema.

Tipos de Algoritmos de Escalonamento

Escalonamento: Tipos de Algoritmos

- Cada ambiente possui requisitos diferentes de escalonamento:
 - Ambiente de Lote: tarefas executadas de forma sequencial.
 - Não interação com o usuário — Preempção não é necessária.
 - Ambiente Interativo (inclui servidores): interação com o usuário.
 - Preempção é necessária
 - Ambiente de Tempo Real: prazos devem ser cumpridos.
 - Processos executados por prioridade — Preempção não é necessária.

Objetivos do Algoritmo de Escalonamento

- Todos os sistemas
 - Justiça — dar a cada processo uma porção justa da CPU.
 - Processos comparáveis devem receber serviços comparáveis.
 - Aplicação da política — verificar se a política estabelecida é cumprida.
 - Exemplo: garantia que a prioridade do processo seja respeitada.
 - Equilíbrio — manter ocupadas todas as partes do sistema.
 - Balanço entre processo limitados por CPU e por E/S para minimizar a competição entre recursos.

Objetivos do Algoritmo de Escalonamento

- Sistemas em lote:
 - Vazão (throughput) — maximizar o número de tarefas por hora.
 - Tempo de retorno — minimizar o tempo entre a submissão e o término.
 - Quanto tempo o usuário médio tem de esperar pela saída.
 - Utilização de CPU — manter a CPU ocupada o tempo todo.
- Observe que Vazão e Tempo de Retorno são objetivos contraditórios.
 - Exemplo: algoritmo que sempre executa as tarefas mais curtas primeiro terá uma alta vazão, porém alto tempo de retorno, pois tarefas longas podem nunca serem selecionadas.

Objetivos do Algoritmo de Escalonamento

- Sistemas Interativos:
 - Tempo de resposta — responder rapidamente às requisições.
 - Tempo entre emitir um comando e receber o resultado.
 - Proporcionalidade — satisfazer às expectativas dos usuários.
 - Usuários têm uma ideia inerente (porém muitas vezes incorreta) de quanto tempo as coisas devem levar.
 - Quando uma solicitação que o usuário percebe como complexa leva muito tempo, os usuários aceitam isso, mas quando uma solicitação percebida como simples leva muito tempo, eles ficam irritados.

Objetivos do Algoritmo de Escalonamento

- Sistemas de Tempo Real:
 - Cumprimento dos prazos — evitar a perda de dados.
 - Previsibilidade — evitar a degradação da qualidade em sistemas multimídia.

Algoritmos de Escalonamento

Escalonamento em Sistemas em Lote

Primeiro a Chegar, Primeiro a Ser Servido

- Não preemptivo.
- Fila de processos prontos.
- Processos bloqueados são colocados no final da fila quando estiverem prontos.
- Fácil compreensão e implementação.
- Tempo de processamento fica limitado pelo processo mais lento.

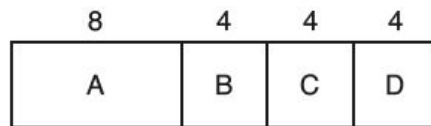
Tarefa mais Curta Primeiro

- Não preemptivo.
- Presume que os tempos de execução são conhecidos antecipadamente.
- As tarefas precisam estar disponíveis simultaneamente.
 - O que ocorre com tempos diferentes de chegada dos processo?
- Produz o tempo de resposta médio mínimo.

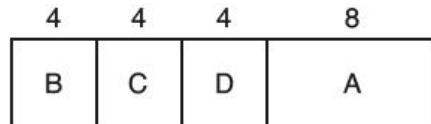
Tarefa mais Curta Primeiro

- Tarefa mais curta primeiro:
 - a) o tempo de retorno para A é 8 minutos, para B é 12 minutos, para C é 16 minutos e para D é 20 minutos, resultando em uma média de 14 minutos.
 - b) tempos de retorno são agora 4, 8, 12 e 20 minutos, resultando em uma média de 11 minutos.

FIGURA 2.41 Um exemplo do escalonamento tarefa mais curta primeiro. (a) Executando quatro tarefas na ordem original. (b) Executando-as na ordem tarefa mais curta primeiro.



(a)



(b)

Tempo mais curto primeiro

- Preemptivo.
- O tempo de execução precisa ser conhecido antecipadamente.
- Escolhe o processo cujo tempo de execução restante é o mais curto.
 - Quando uma nova tarefa chega, seu tempo total é comparado com o tempo restante do processo atual;
 - Se a nova tarefa precisa de menos tempo para terminar do que o processo atual, este é suspenso e a nova tarefa iniciada;

Escalonamento em Sistemas Interativos

Escalonamento por Chaveamento Circular

- Circular (round-robin): A cada processo é designado um intervalo, chamado de seu quantum, durante o qual ele é deixado executar.
 - Se o processo ainda está executando ao fim do quantum, a CPU sofrerá uma preempção e receberá outro processo.
 - Se o processo foi bloqueado ou terminado antes de o quantum ter decorrido, o chaveamento de CPU será feito quando o processo bloquear, é claro.
- Fácil implementação.
- Como determinar o tamanho do quantum?

Escalonamento por Chaveamento Circular

- Considere um quantum de 4 ms, com tempo de 1 ms para chaveamento dos processos.
 - Overhead de 20% do processamento.
- Considere um quantum de 100 ms, agora.
 - Overhead de 1% do processamento.
 - Considere 50 processos, o último processo na fila deverá esperar 5 segundos para ser executado.

Escalonamento por Chaveamento Circular

- Se o quantum for mais longo que o surto de CPU médio, a preempção não acontecerá com muita frequência.
 - Bloqueio antes de encerrar o quantum.
 - Escalonamento não preemptivo terá melhor desempenho.

Escalonamento por Chaveamento Circular

- Quantum curto demais provoca muitos chaveamentos de processos e reduz a eficiência da CPU.
- Quantum longo demais pode provocar uma resposta ruim a solicitações interativas curtas.
- Prática: quantum em torno de 20-50 ms é muitas vezes bastante razoável.

Escalonamento por Prioridades

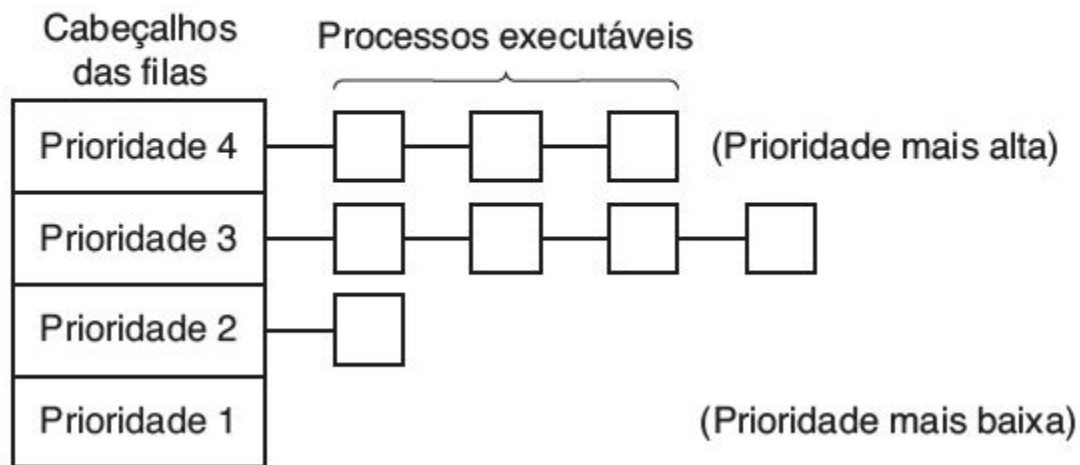
- A cada processo é designada uma prioridade, e o processo executável com a prioridade mais alta é autorizado a executar.
- Como evitar que processos de alta prioridade executem indefinidamente?
 - Decremento da prioridade a cada interrupção de relógio.
 - Quantum de tempo máximo.

Escalonamento por Prioridades

- Como determinar as prioridades?
- Prioridades estáticas: são fixas durante a vida do processo.
- Prioridades dinâmicas: mudam ao longo do tempo.
 - Exemplo: $1/f$ onde f é a fração do último quantum que o processo usou.
 - Favorece processo IO-bound, pois eles devem receber a CPU imediatamente para efetuarem a requisição de E/S, voltando ao estado de bloqueado.
- Classes de prioridade.
 - Escalonamento de prioridade entre as classes.
 - Escalonamento circular dentro de cada classe.

Escalonamento por Prioridades

FIGURA 2.43 Um algoritmo de escalonamento com quatro classes de prioridade.



Processo mais Curto em Seguida

- Como estimar o tempo dos processos?
- Estimação baseada nos tempo passados.
- Processos interativos geralmente seguem o padrão de esperar pelo comando, executar o comando, esperar pelo comando, executar o comando etc.

Processo mais Curto em Seguida

- Suponha o tempo estimado por comando para alguns processos é T_0 .
- Suponha que a execução seguinte é mensurada como sendo T_1 .
- Técnica de aging (envelhecimento): estimativa atualizada pela média ponderada.
 - $a \cdot T_0 + (1 - a) \cdot T_1$.
 - Constante a determina a importância das estimativas passadas.

Política versus Mecanismo

Política versus Mecanismo

- Em grupos de processo, o processo pai pode saber qual dos filhos é mais importante.
- Os algoritmos vistos não possuem acesso a essa informação.
 - O escalonador raramente faz a melhor escolha.
- Separação:
 - Mecanismo de escalonamento.
 - Política de escalonamento.

Política versus Mecanismo

O que isso significa é que o algoritmo de escalonamento é parametrizado de alguma maneira, mas os parâmetros podem estar preenchidos pelos processos dos usuários.

Resumo

Resumo

- Escalonamento e Algoritmo de Escalonamento
- Processo CPU-bound e IO-bound
- Quando ocorre o escalonamento
- Custo do escalonamento
- Objetivos dos algoritmos de escalonamento
- Política versus Mecanismo

Resumo

- Algoritmos
 - Primeiro a Chegar, Primeiro a Ser Servido
 - Tarefa mais Curta Primeiro
 - Tempo mais Curto Primeiro
 - Chaveamento Circular
 - Prioridades
 - Processo mais Curto Primeiro

Dúvidas?

