



e-Learning



Research



www.professorlima.com

Aula Anterior

Gerenciamento de
Processos - Parte III
Threads
Modelos de Criação de
Threads

Aula de Hoje

Gerenciamento de
Processos
Escalonamento (FIFO e
Prioridade)
Troca de Contexto

Próxima Aula

Escalonamento de
Processos
SJF-Preemptivo e Não-
Preemptivo
Round Robin rou
Alternância Circular

Cronograma

Escalonamento de Processos

- O escalonamento de processadores é a forma com os processadores existentes num sistema computacional são utilizados para efetuar o processamento, isto é, é como os processos são distribuídos para execução nos processadores.

Tanenbau propõe a seguinte definição

Quando mais de um processo é executável, o sistema operacional deve decidir qual será executado primeiro. A parte do sistema operacional dedicada a esta decisão é chamada escalonador (scheduler) e o algoritmo utilizado é chamado algoritmo de escalonamento (scheduling algorithm). [TAN92, p. 62]

Escalonamento de Processos

Por sua vez, Deitel coloca que:

A designação de processadores físicos para processos permite aos processos a realização de trabalho. Esta designação é uma tarefa complexa realizada pelo sistema operacional. Isto é chamado escalonamento do processador (processor scheduling).

Escalonamento de Processos

- Simplificando, podemos afirmar que num sistema onde só exista um único processador, o escalonamento representa a ordem em que os processos serão executados.
- A forma com que se dá o escalonamento é, em grande parte, responsável pela produtividade e eficiência atingidas por um sistema computacional. Mais do que um simples mecanismo, o escalonamento deve representar uma política de tratamento dos processos que permita obter os melhores resultados possíveis num sistema.

Objetivos do escalonamento

O projeto de um escalonador adequado deve levar em conta uma série de diferentes necessidades, ou seja, o projeto de uma política de escalonamento deve contemplar os seguintes objetivos:

- Ser justo: todos os processos devem ser tratados igualmente, tendo possibilidades idênticas de uso do processador, devendo ser evitado o adiamento indefinido.
- Maximizar a produtividade (throughput): procurar maximizar o número de tarefas processadas por unidade de tempo.
- Ser previsível: uma tarefa deveria ser sempre executada com aproximadamente o mesmo tempo e custo computacional.

Objetivos do escalonamento

- Minimizar o tempo de resposta para usuários interativos.
- Maximizar o numero possível de usuários interativos.
- Minimizar a sobrecarga (overhead): recursos não devem ser desperdiçados embora algum investimento em termos de recursos para o sistema pode permitir maior eficiência.

Objetivos do escalonamento

- Favorecer processos bem comportados: processos que tenham comportamento adequado poderiam receber um serviço melhor.
- Balancear o uso de recursos: o escalonador deve manter todos os recursos ocupados, ou seja, processos que usam recursos subutilizados deveriam ser favorecidos.
- Exibir degradação previsível e progressiva em situações de intensa carga de trabalho.

Objetivos do escalonamento

- Como pode ser visto facilmente, alguns destes objetivos são contraditórios, pois dado que a quantidade de tempo disponível de processamento (tempo do processador) é finita, assim como os demais recursos computacionais, para que um processo seja favorecido outro deve ser prejudicado.
- O maior problema existente no projeto de algoritmos de escalonamento está associado a natureza imprevisível dos processos, pois não é possível prevermos se um dado processo utilizará intensamente o processador, ou se precisará grandes quantidades de memória ou se necessitará numerosos acessos aos dispositivos de E/S.

- Existem três níveis distintos de escalonamento em um sistema computacional quando se considera a frequência e complexidade das operações envolvidas.

Escalonamento de alto nível

- Chamado também de escalonamento de tarefas, corresponde a admissão de processos, isto é, a determinação de quais tarefas passarão a competir pelos recursos do sistema. Uma vez admitidas, as tarefas transformam-se em processos. Correspondem a rotinas de alto nível oferecidas pelas APIs do sistema operacional.

Níveis de escalonamento

Escalonamento de nível intermediário

- Corresponde a determinação de quais processos existentes competirão pelo uso do processador (processos ativos). Este nível de escalonamento é responsável por administrar a carga do sistema, utilizando-se de primitivas de suspensão (suspend) e ativação (resume ou activate). Correspondem a rotinas internas do sistema operacional.

Níveis de
escalonamento

Escalonamento de baixo nível

- Rotinas que determinam qual processos, dentre os processos ativos, será o próximo processo que efetivamente utilizará o processador. Estas tarefa são executadas pelo dispatcher, usualmente uma rotina escrita diretamente em linguagem de m'equina que se encontra permanentemente na memória principal.

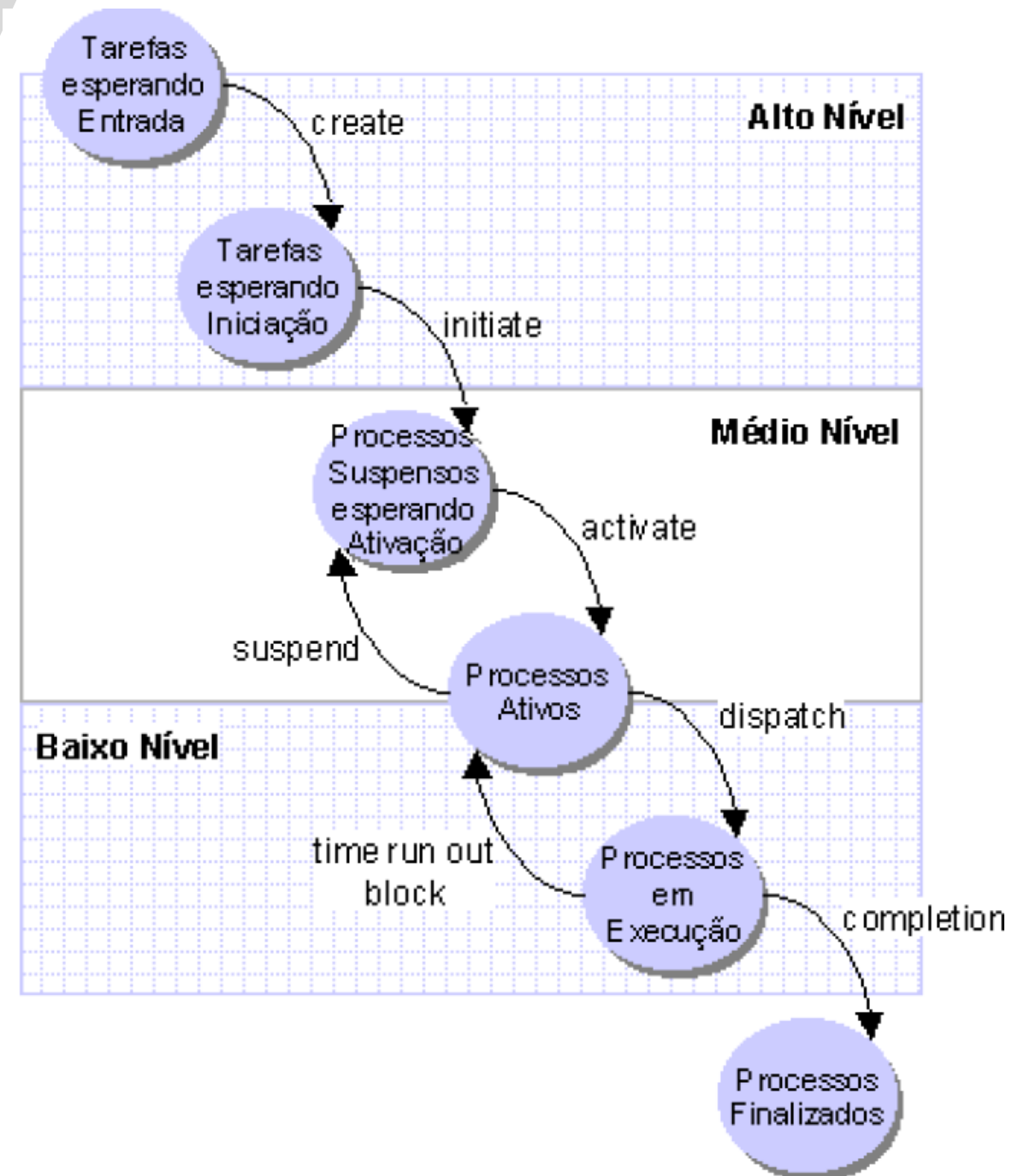
Níveis de
escalonamento

- Os níveis de escalonamento alto, intermediário e baixo também são conhecidos respectivamente como escalonamento de longo prazo, médio prazo e curto prazo. O escalonamento de alto nível ou de longo prazo ocorre menos frequentemente num sistema enquanto o escalonamento de baixo nível ou de curto prazo ocorre constantemente, dado que representa a troca de contexto e o chaveamento do processador entre os processos ativos.

Níveis de escalonamento

Níveis de escalonamento

- Considerando assim os níveis de escalonamento e as operações de suspensão (suspend ou sleep) e ativação (resume, wakeup ou activate), o mapa de estados dos processos pode ser representado de maneira mais completa



Escalonamento preemptivo e não preemptivo

Um algoritmo de escalonamento é dito **não preemptivo** quando temos que o processador designado para um certo processo não pode ser retirado deste até que o processo seja finalizado (completion).

1

Analogamente, um algoritmo de escalonamento é considerado **preemptivo** quando o processador designado para um processo pode ser retirado deste em favor de um outro processo.

Escalonamento preemptivo e não preemptivo

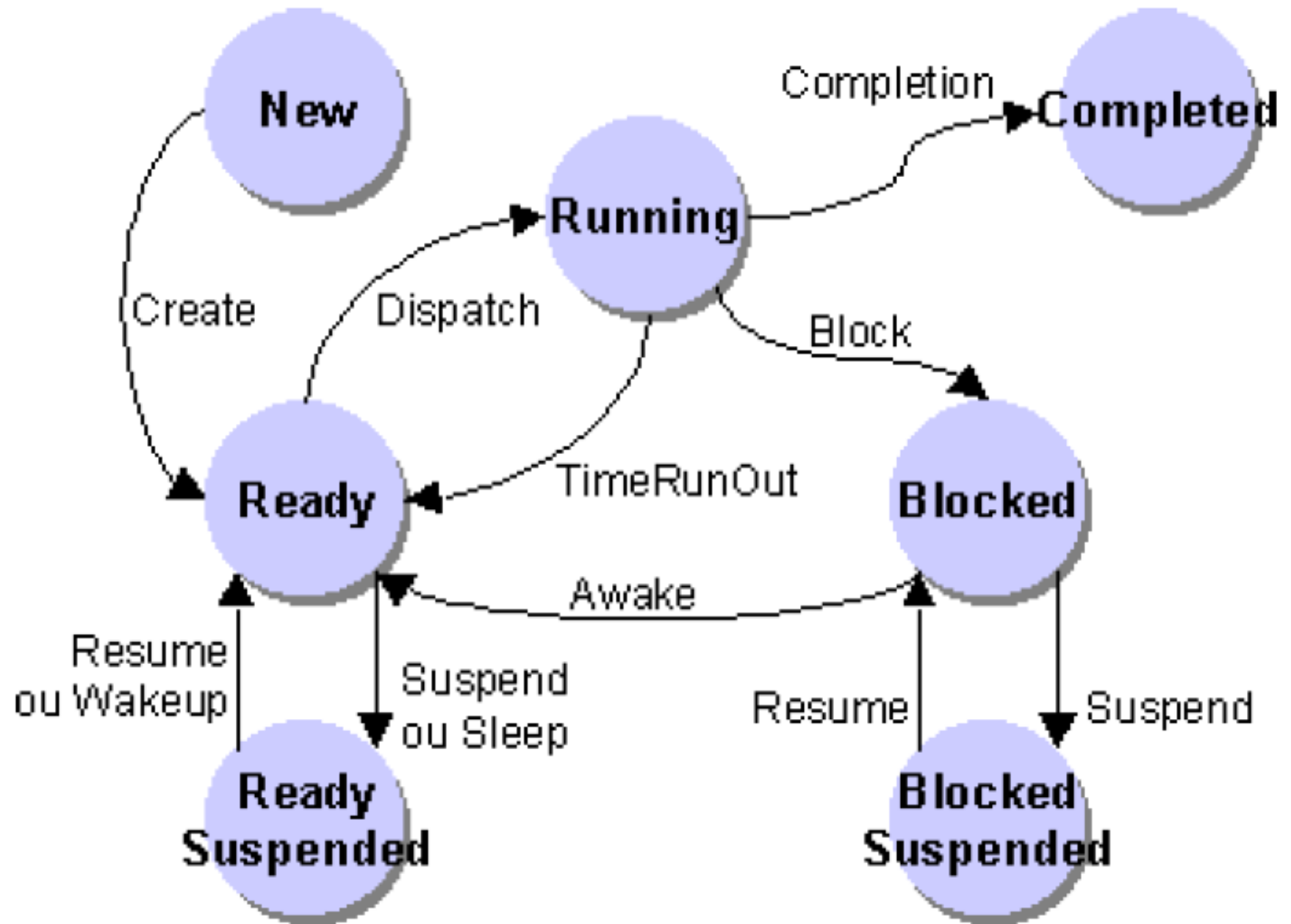
- Algoritmos preemptivos são mais adequados para sistemas em que múltiplos processos requerem atenção do sistema, ou seja, no caso de sistemas multiusuário interativos (sistemas em tempo repartido) ou em sistema de tempo real.
- Nestes casos, a preemptividade representa a troca do processo em execução, assim sendo, para que o processador seja retirado de um processo, interrompendo seu trabalho, e designado a outro processo, anteriormente interrompido, é fundamental que ocorra a troca de contexto dos processos.

Escalonamento preemptivo e não preemptivo

Tal troca exige que todo o estado de execução de um processo seja adequadamente armazenado para sua posterior recuperação, representando uma sobrecarga computacional para realização desta troca e armazenagem de tais dados. Usualmente os algoritmos preemptivos são mais complexos dada a natureza imprevisível dos processos.

Por sua vez, os algoritmos não preemptivos são mais simples e adequados para o processamento não interativo, semelhante aos esquemas de processamento em lote dos sistemas batch. Embora não proporcionando interatividade, são geralmente mais eficientes e previsíveis quanto ao tempo de entrega de suas tarefas.

Escalonamento
preemptivo e
não
preemptivo



Escalonamento preemptivo e não preemptivo

- Existem também algoritmos de escalonamento cooperativo, onde os processos não são interrompidos, mas a preempção ocorre em duas situações bem definidas: quando o processo efetua uma operação de I/O e quando o processo é finalizado.
- Também é possível que um processo ceda o processador, voluntariamente, em favor de outros processos. Neste caso, o processo educado poderia executar uma chamada a uma função yield (dar preferencia), como no caso do MS-Windows 3.1.
- A preempção voluntária pode auxiliar numa distribuição mais equitativa da capacidade de processamento do sistema, mas conta com a generosidade do programador, nem sempre disponível.

Escalonamento preemptivo e não preemptivo

- A preemptividade de certos algoritmos se baseia no fato de que o processador é, naturalmente, um recurso preemptivo, ou seja, um recurso que pode ser retirado de um processo e posteriormente devolvido sem prejuízo.
- O mesmo acontece com a memória. Por outro lado, outros tipos de recursos não podem sofrer preempção, tais como impressoras e até mesmo arquivos, dado que muitas vezes não podem ser retirados de um processo sem que ocorra prejuízo para este.

“Seja o que a sua mente pode
conceber e acreditar, ela pode
conseguir”

Napoleon Hill



Thank you

devel-it.slack.com
Telegram 11 9 5850 8384
E-mail: lima@clips.com.br



ProfessorLima.Com