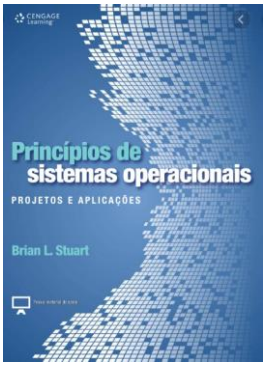
FACULDADE DE COMPUTAÇÃO E INFORMÁTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO
SISTEMAS OPERACIONAIS – Aula 03 – 2º SEMESTRE/2020
PROF. LUCIANO SILVA

TEORIA: ESCALONAMENTO DE PROCESSOS

Nossos objetivos nesta aula são:

- aprofundar o estudo do mecanismo de escalonamento de processos em sistemas operacionais
- conhecer e avaliar os principais algoritmos de escalonamento de processos
- conhecer a implementação de escalonamento do MINIX



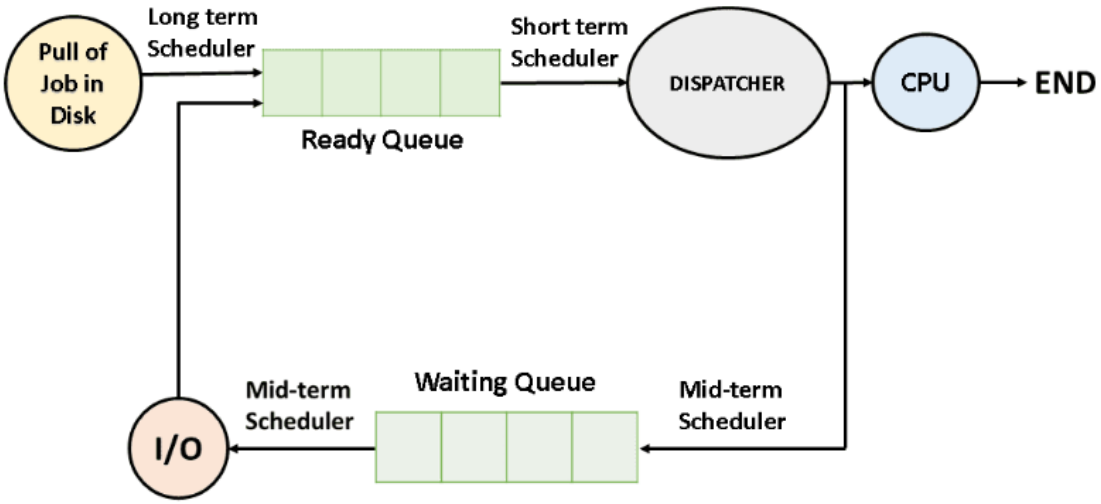
Para esta aula, usamos como referência a **Seção 5.4** do nosso livro-texto:

STUART, B.L**. Princípios de Sistemas Operacionais: Projetos e Aplicações**. São Paulo: Cengage Learning, 2011.

Não deixem de ler esta seção depois desta aula!

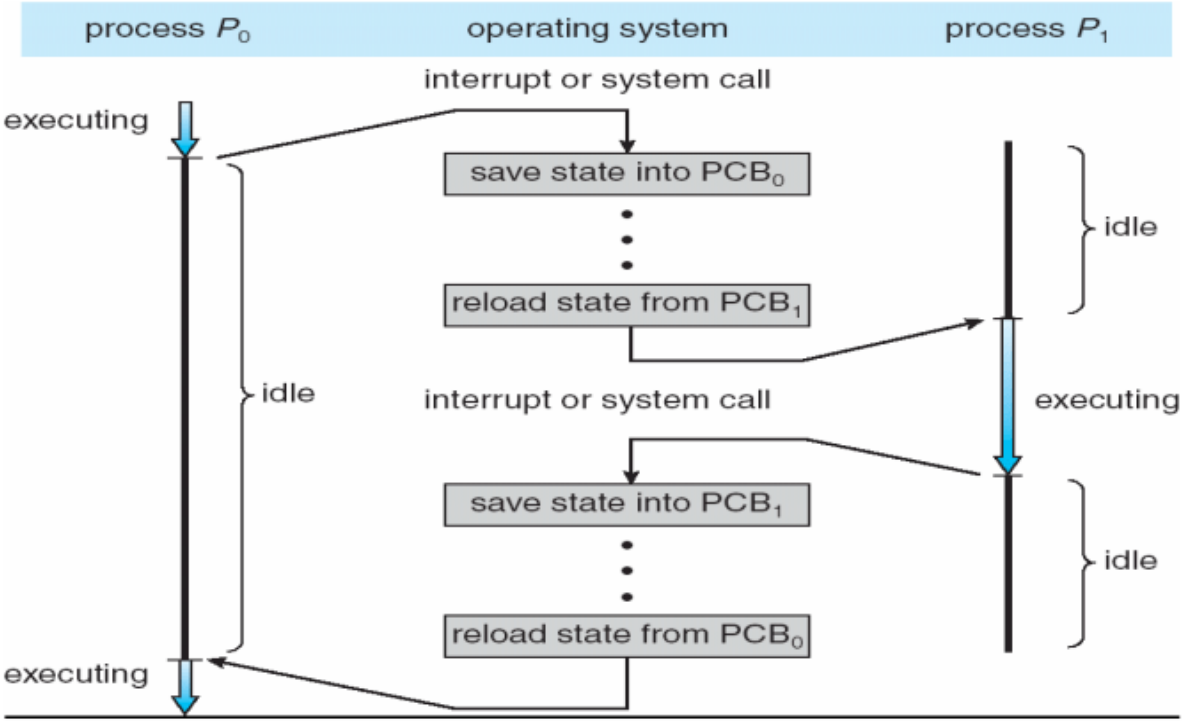
ESCALONAMENTO DE PROCESSOS

- O **escalonamento (*scheduling*) mais comum** é o mecanismo pelo qual o S.O. **seleciona qual processo no estado de PRONTO irá obter o uso de CPU** e passar para o estado de **RODANDO**. Este escalonador também é chamado de **escalonador de cadência curta ou escalonador de CPU *(short term scheduler*)**.



- Existem outros escalonadores de cadência menor: **mid-term scheduler**, que provoca a mudança de contexto de RODANDO para SUSPENSO/de SUSPENSO para PRONTO e o **long term scheduler (escalonador de jobs)**, que transforma os jobs em processos prontos.

- O **dispatcher** é o responsável por realizar a **mudança de contexto (context switching)**, uma vez que o escalonador tenha decidido qual processo irá utilizar a CPU.



EXERCÍCIO COM DISCUSSÃO EM DUPLAS

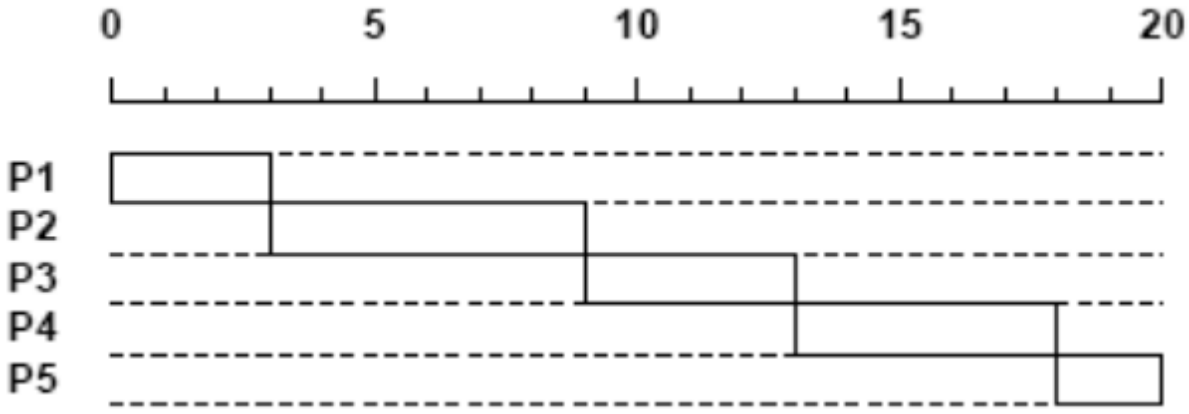
Geralmente, um escalonador de CPU precisa ser muito mais rápido do que um escalonador de jobs ou escalonador de cadência média (mid-term scheduler). Explique por quê.

ALGORITMOS DE ESCALONAMENTO

- O algoritmo mais elementar de escalonamento de processos é o **FCFS (First-Come First-Served)**, **PRIMEIRO QUE CHEGA-PRIMEIRO A SER ATENDIDO**. A implementação deste algoritmo é bastante simples: basta uma fila simples para gerenciar os processos. Uma vez que o processo ganhe o uso da CPU, aguarda-se seu término para pegar o próximo da fila.

- Abaixo, temos um exemplo de execução deste algoritmo:

Processo	Tempo de chegada	Tempo de serviço
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2



- Para avaliar um algoritmo de escalonamento, podemos utilizar o **tempo de vida para cada processo** e o **tempo de vida médio**. O tempo de vida para um processo i é o tempo total em que entre o momento da submissão do processo (tempo de chegada) e o momento em que este processo esteja finalizado. O tempo de vida médio é a média aritmética simples dos tempos de vida para cada processo.

EXERCÍCIO TUTORIADO

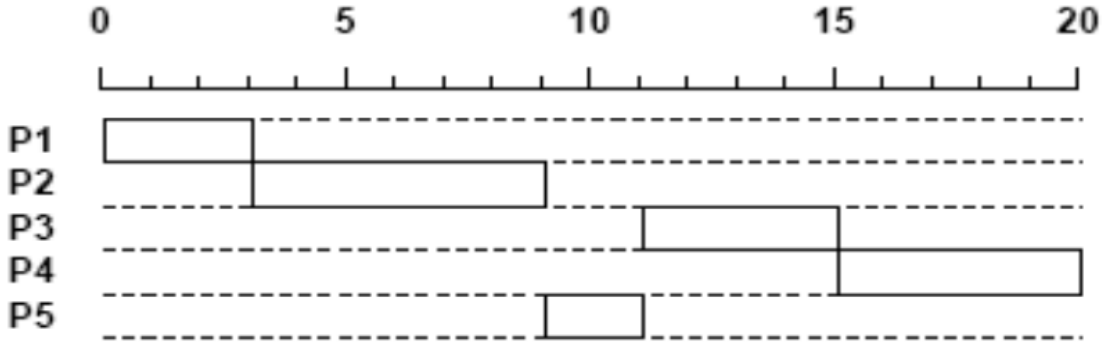
Calcule o tempo de vida para cada um dos processos abaixo e o tempo de vida médio, considerando o algoritmo FCFS:

Processo	Tempo de chegada	Tempo de serviço
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

ALGORITMOS DE ESCALONAMENTO (Continuação)

- A segunda estratégia de escalonamento é a **SJF (Shortest Job First) (PROCESSO MAIS CURTO PRIMEIRO).** Neste esquema, o processo que declarar o menor tempo de utilização da CPU e estiver disponível para escalonamento, será o escolhido. Abaixo, temos um exemplo de escalonamento SJF:

Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2



- Uma variação deste algoritmo é o **PRÓXIMO DE MENOR TEMPO RESTANTE**. Neste algoritmo, será escalonado o processo disponível cujo **tempo de término seja o menor possível**.

EXERCÍCIO COM DISCUSSÃO EM DUPLAS

Calcule o tempo de vida para cada um dos processos abaixo e o tempo de vida médio, considerando o algoritmo SJF:

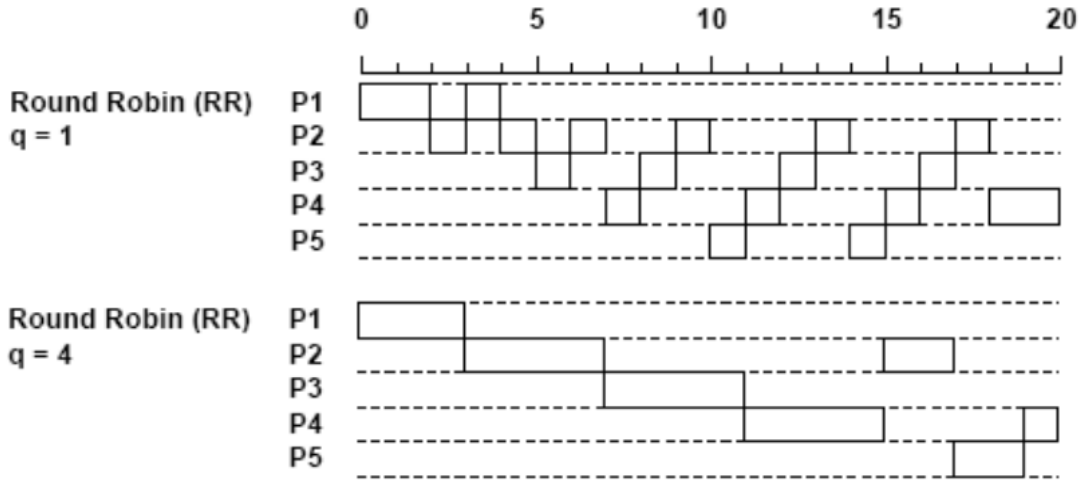
Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2

ALGORITMOS DE ESCALONAMENTO (Continuação)

- A terceira estratégia de escalonamento é a **RR (Round Robin)**, que é um escalonamento com **fila circular** e onde todos os **processos possuem a mesma fatia de tempo (quantum)** para usar a CPU. Caso o processo não termine nesta fatia de tempo, ele volta para a fila de processos prontos para esperar a próxima rodada de execução.

- Abaixo, temos um exemplo de escalonamento RR com dois diferentes quantuns de tempo. Sistemas operacionais atuais usam, normalmente, um quantum variando de 10ms a 100ms.

Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2



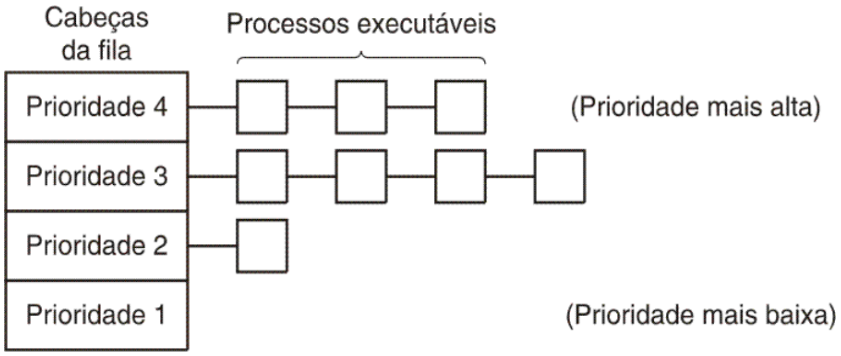
EXERCÍCIO COM DISCUSSÃO EM DUPLAS

Calcule o tempo de vida para cada um dos processos abaixo e o tempo de vida médio, considerando o algoritmo RR com quantum q=1.

Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2

ALGORITMOS DE ESCALONAMENTO (Continuação)

- Finalmente, a quarta estratégia possível inclui **prioridades**. Processos com prioridade mais alta possuem probabilidade de serem escalonados mais vezes. Uma maneira bastante simples de controle de processos com prioridade é com a utilização de múltiplas filas de processos, uma para cada prioridade.



EXERCÍCIO COM DISCUSSÃO EM DUPLAS

Um problema sério que pode ocorrer no esquema de escalonamento por prioridade é o fenômeno de **starvation (inanição)**. Se sempre houver processos de maior prioridade a serem escalonados, os processos de menor prioridade correm o risco de nunca usarem a CPU, pelo fato de nunca serem escalonados.

Proponha uma estratégia para evitar a ocorrência de starvation.

EXERCÍCIOS EXTRA-CLASSE

1. Calcule o tempo de vida para cada um dos processos abaixo e o tempo de vida médio, considerando o algoritmo RR (Round Robin) com quantum q=4.

Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2

2. Considere novamente a seguinte tabela de processo:

Processo	Tempo de chegada	Tempo de serviço
P1	0	3
P2	2	6
P3	4	4
P4	6	5
P5	8	2

   1. Calcule o tempo de vida de cada processo pelas técnicas FCFS, SJF e RR (quantum = 5s).
   2. Calcule o tempo de vida médio dos processos pelas técnicas FCFS, SJF e RR (quantum = 5s).
   3. Compare os três algoritmos (FCFS, SJF e RR) segundo o tempo de vida médio de cada um dos processos.

3. Defina o funcionamento dos seguintes algoritmos adicionais para escalonamento:

   - Filas de retorno multinível
   - Inversão de prioridade
   - Escalonamento de dois níveis
   - Escalonamento guiado por eventos
   - Escalonamento de slots

4. A figura abaixo mostrar uma maneira alternativa de se implementar a comunicação entre a fila de processos prontos e a fila de processo suspensos. Argumente por que esta modificação poderia ficar mais eficiente.