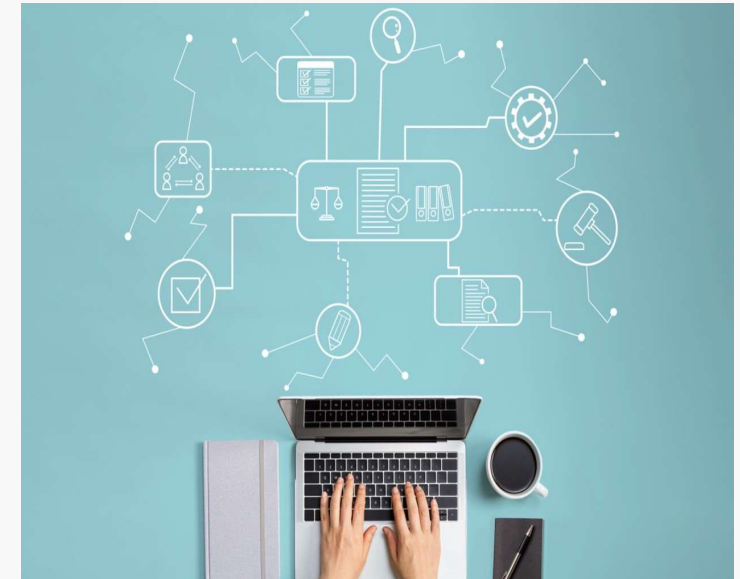


GERENCIAMENTO DE PROCESSOS

GRUPO: AMANDA MIKELY ABREU
MACEDO, EMANUELLE DA SILVA LAUNE,
GABRIELA TORRES DE QUEIROZ



Sumário



1. Conceitos Básicos



2. Categorias de Escalonamento de Processos



3. Sistemas Batch



4. Sistemas Interativos



5. Sistemas de Tempo Real



6. Referências

1. Conceitos Básicos



Sistema Operacional é um conjunto de ferramentas que possibilita o uso adequado do computador.



Processo é um programa em execução.



Escalonamento é um processo responsável por escolher os processos que serão executados pelo processador.

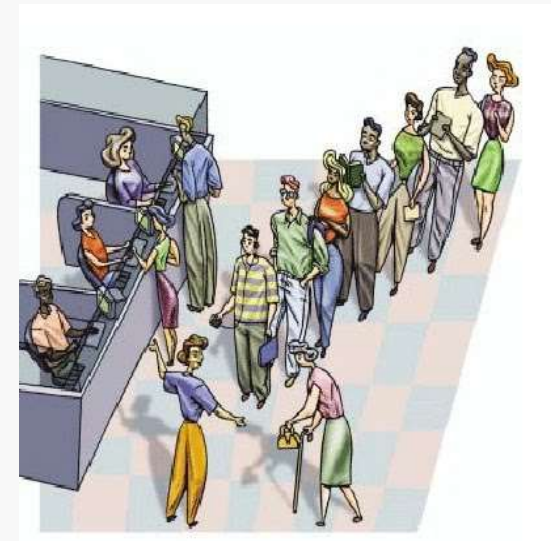
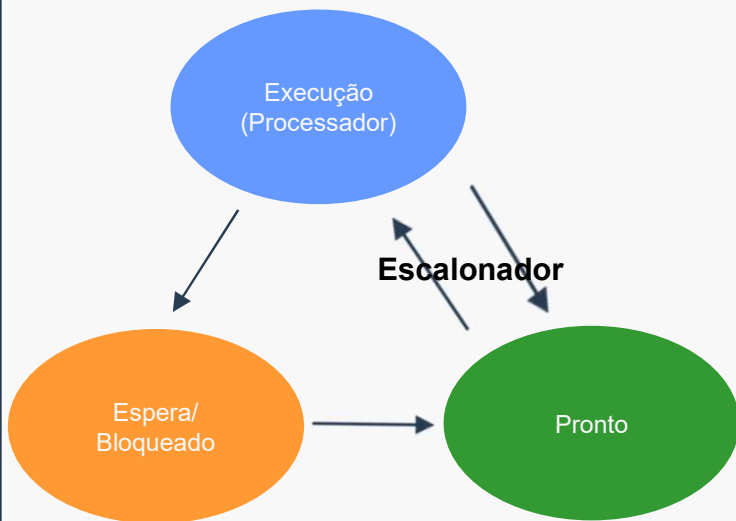


Figura 1: Fila do banco

1. Conceitos Básicos



Mudança de contexto

- É a mudança de estados;
- Exemplo: quando saio de um estado de execução e vou para um estado de bloqueado.



Dispatcher

- Responsável pelo o que acontece na mudança de contexto.




Escalonador

- É quem vai escolher a tarefa que vai fazer uso do processador.



1. Conceitos Básicos



- O Escalonador de Processos escolhe o processo que será executado pela CPU seguindo os seguintes critérios:
 - Utilização do processador ;
 - Throughput;
 - Tempo de Processador;
 - Tempo de Espera;
 - Tempo de Turnaround;
 - Tempo de Resposta.
- 

1. Conceitos Básicos



Tipos de escalonadores

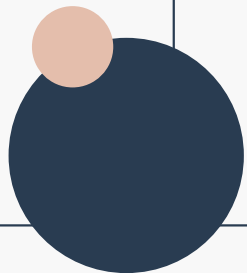
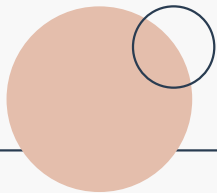
Preemptivo

- Provoca interrupção forçada para que outro processo possa utilizar a CPU;
- Parar de fazer o bolo e ajuda a criança.



Não-preemptivo

- Não interrompe a execução de um processo;
- Continua fazendo o bolo e deixa a criança chorando.



2. Categorias

2.Categorias

- Sistemas Batch:** usuários não esperam por respostas rápidas;
- Sistemas Interativos:** interação constante do usuário;
- Sistemas de Tempo Real:** processos são executados mais rapidamente e o tempo é o fator crucial.

3. Sistemas Batch

- Sistema onde descarrego um conjunto de programas para serem executados
- O processamento batch tem a característica de não exigir a interação do usuário com a aplicação. Todas as entradas e saídas de dados da aplicação são implementadas por algum tipo de memória secundária, geralmente arquivos em disco.

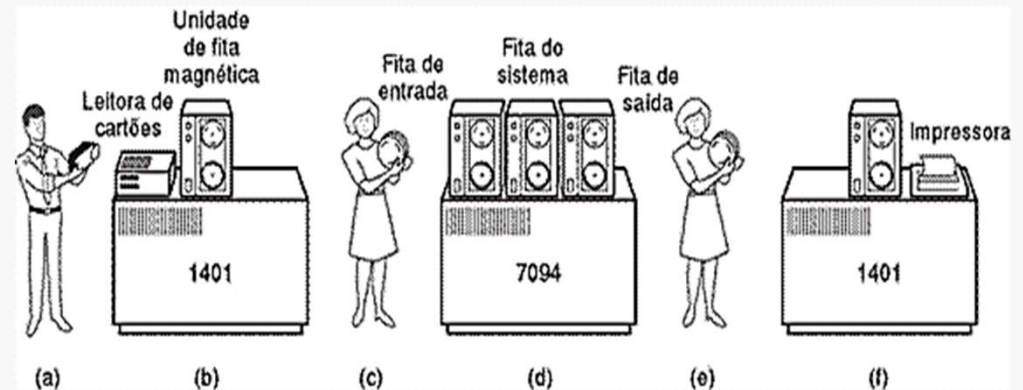


Figura 2: Sistemas Batch

3. Sistemas Batch

Algoritmos para Sistemas em Batch:

First-come
first-serve
(FIFO)

Shortest Job
First

Shortest
Remaining
Time Next

3. Sistemas Batch – FIFO



Não-preemptivo;



Processos são executados na CPU seguindo a ordem de requisição;



Fácil de entender e programar;



Maior turnaround;



Desvantagem: Ineficiente quando se tem processos que demoram na sua execução.

3. Sistemas Batch - FIFO

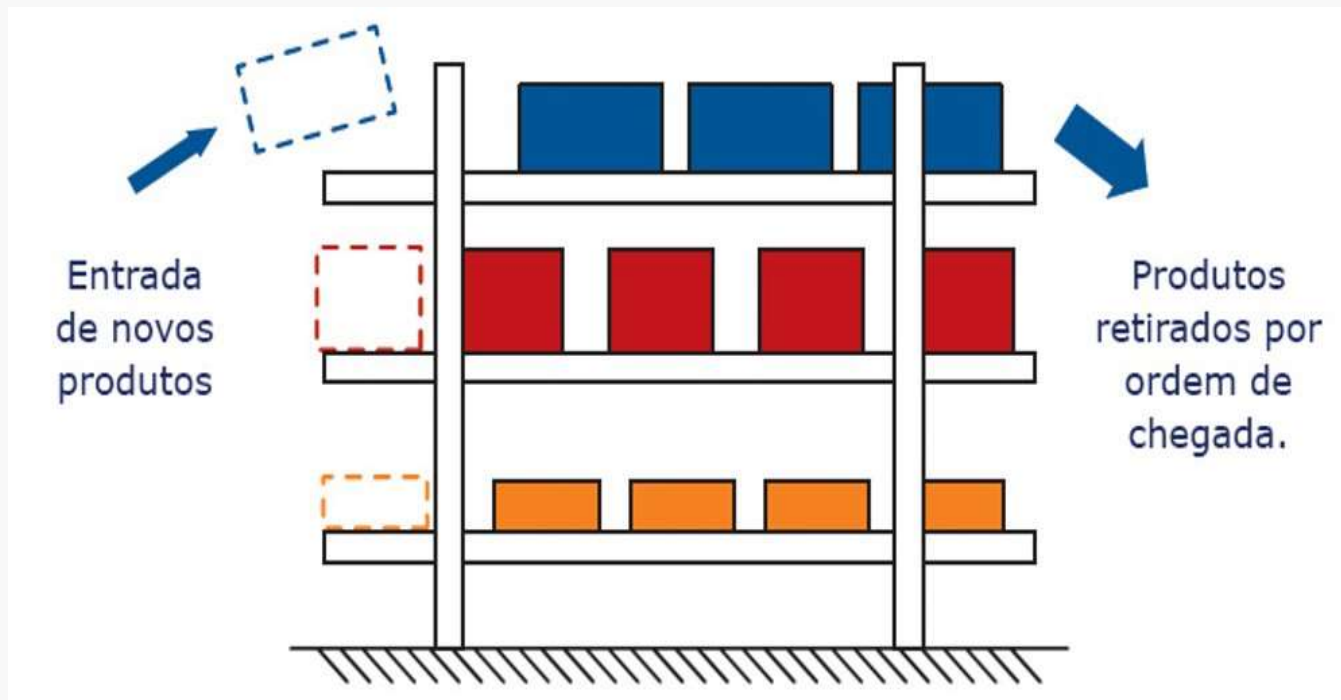


Figura 3: Representação do sistema FIFO

3. Sistemas Batch - FIFO

- Na imagem ao lado temos a exemplificação do algoritmo FIFO, a ordem de chegada é o único critério de escalonamento.

8	4	4	4
A	B	C	D

Em ordem:

Turnaround A = 8

Turnaround B = 12

Turnaround C = 16

Turnaround D = 20

Média $\rightarrow 56/4 = 14$

Figura 4: Exemplo do algoritmo FIFO



Não-preemptivo;



Possível prever o tempo de execução do processo;



Menor processo é executado primeiro;



Menor turnaround;



Desvantagem: baixo aproveitamento quando se tem poucos processos prontos para serem executados;

3. Sistemas Batch – Shortest Job First (SJF)

3. Sistemas Batch - SJF

8	4	4	4
A	B	C	D

Em ordem:

Turnaround A = 8

Turnaround B = 12

Turnaround C = 16

Turnaround D = 20

Média $\rightarrow 56/4 = 14$

4	4	4	8
B	C	D	A

Menor *job* primeiro:

Turnaround B = 4

Turnaround C = 8

Turnaround D = 12

Turnaround A = 20

Média $\rightarrow 44/4 = 11$

Figura 5: Exemplo algoritmo SJF

- Na imagem acima temos a comparação entre o algoritmo FIFO e o algoritmo SJF, percebe-se que o tempo de turnaround do SJF é menor.

3. Sistemas Batch – Shortest Remaining Time Next (SRTN)



Preemptivo;



Possível prever o tempo de execução do processo;



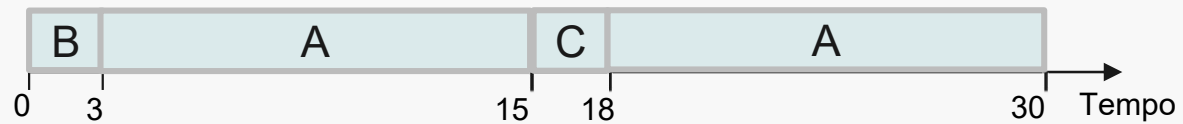
Menor processo é executado primeiro;



Desvantagem: processos que consomem mais tempo podem demorar muito para serem finalizados.

3. Sistemas Batch – Shortest Remaining Time Next (SRTN)

Processo:	A	B	C
Duração:	24	3	3
Tempo de espera:	$3+3=6$	0	0





4. Sistemas Interativos

- São sistemas que consistem em dividir o tempo de processamento entre vários usuários;
- Usuários interagem diretamente com o sistema;
- Escalonadores preemptivos são essenciais.

4. Sistemas Interativos

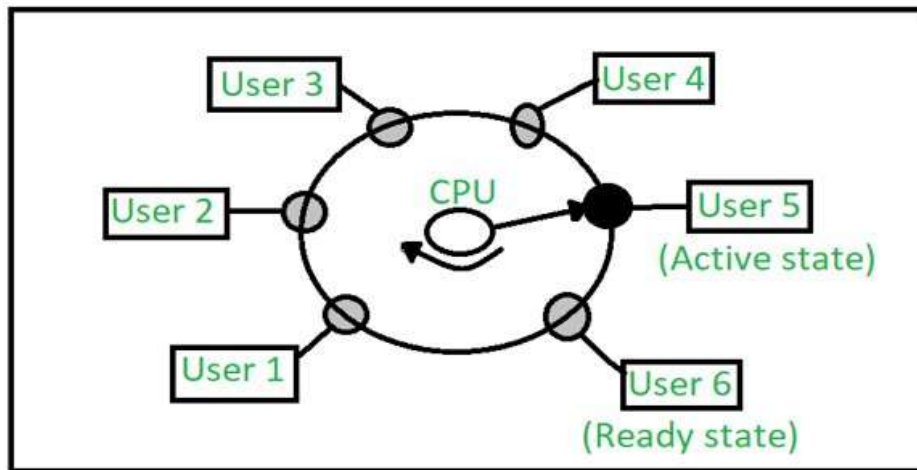


Figura 6: Exemplo de sistemas interativos

Vantagens:

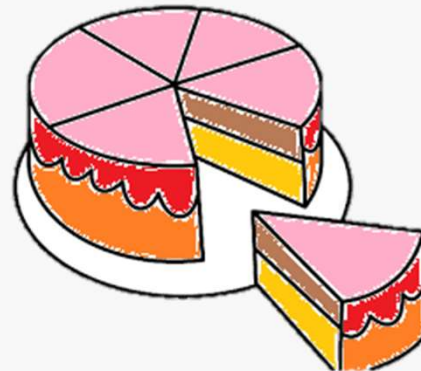
- Experiência de usuário aprimorada: o compartilhamento de tempo fornece um ambiente interativo que permite aos usuários se comunicarem com o computador em tempo real;
- Oferece a vantagem de uma resposta rápida;
- Reduz o tempo ocioso da CPU.

Desvantagens:

- Alta sobrecarga: os sistemas de compartilhamento de tempo têm uma sobrecarga maior do que outros sistemas operacionais devido à necessidade de agendamento, alternância de contexto e outras sobrecargas que acompanham o suporte a vários usuários;
- É preciso cuidar da segurança e integridade dos programas e dados do usuário.

4. Sistemas Interativos – Round Robin Scheduling

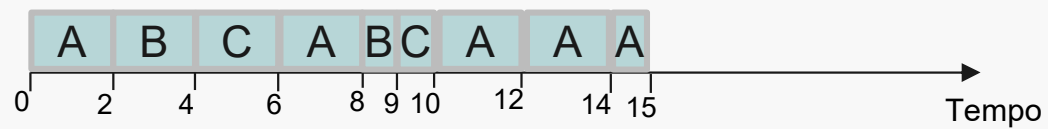
- Preemptivo;
- Cada processo recebe um tempo de execução chamado quantum; ao final desse tempo, o processo é suspenso e outro processo é colocado em execução;
- Quantum: se for muito pequeno, ocorrem muitas trocas diminuindo, assim, a eficiência da CPU; se for muito longo o tempo de resposta é comprometido.



. Sistemas Interativos – Round Robin Scheduling

Quantum de 2s

Processo:	A	B	C
Duração:	9	3	3





Cada processo possui uma prioridade → os processos prontos com maior prioridade são executados primeiro;



Prioridades são atribuídas dinâmica ou estaticamente;



Classes de processos com mesma prioridade;



Preemptivo;

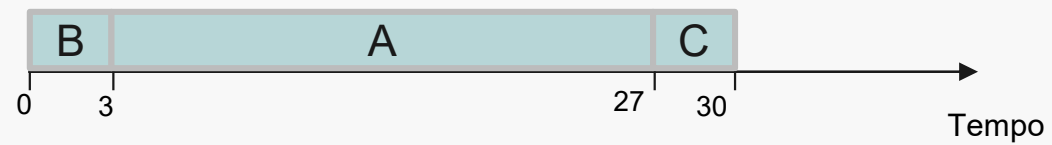


Cada processo possui um quantum.

4. Sistemas Interativos – Por Prioridade

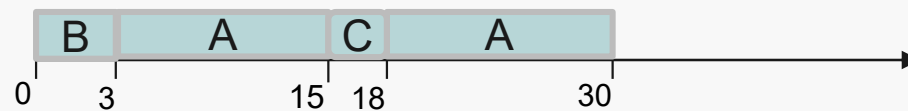
4. Sistemas Interativos – Por Prioridade

Processos:	A	B	C
Duração:	24	3	3
Prioridade:	1	0	2



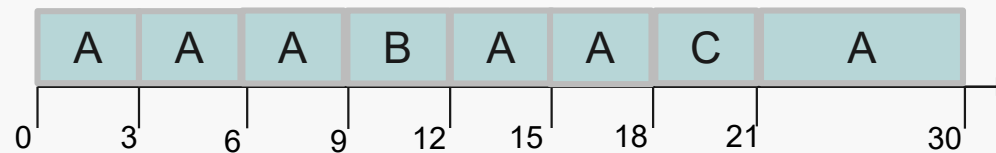
4. Sistemas Interativos – Por Prioridade com preempção

Processos:	A	B	C
Duração:	24	3	3
Prioridade:	1	0	0



4. Sistemas Interativos – Por Prioridade

Processos:	A	B	C
Duração:	24	3	3
Prioridade:	0	1	0



4. Sistemas Interativos - Múltiplas Filas



Os processos podem ser divididos em diferentes filas, dependendo da sua prioridade;



Classes de prioridades;



Cada classe de prioridades possui quantum diferentes;



Cada vez que um processo é executado e suspenso ele recebe mais tempo para execução;



Preemptivo.

Algoritmo de Múltiplas Filas

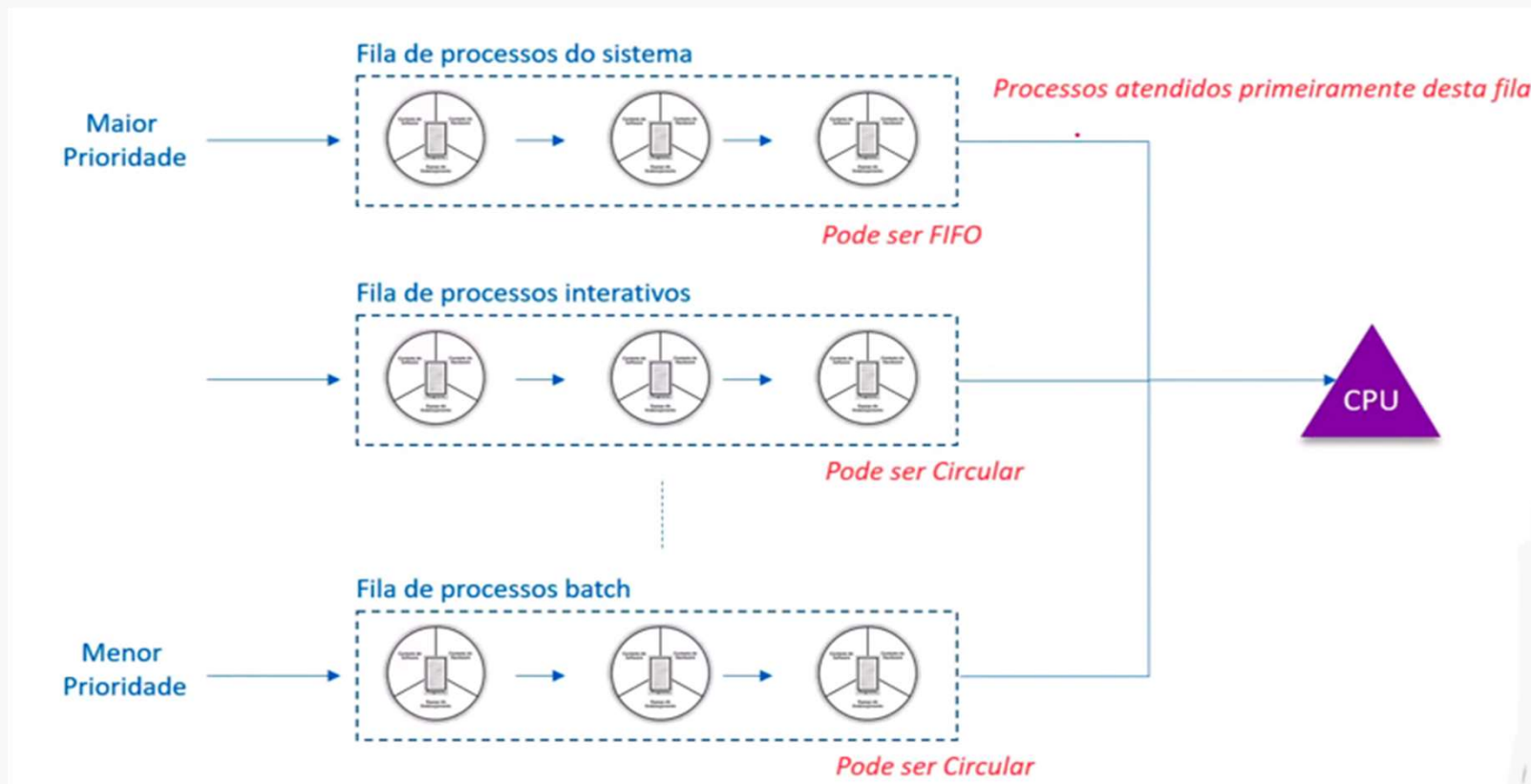


Figura 7: Exemplo de algoritmo de múltiplas filas

Algoritmo de Múltiplas Filas - Realimentação

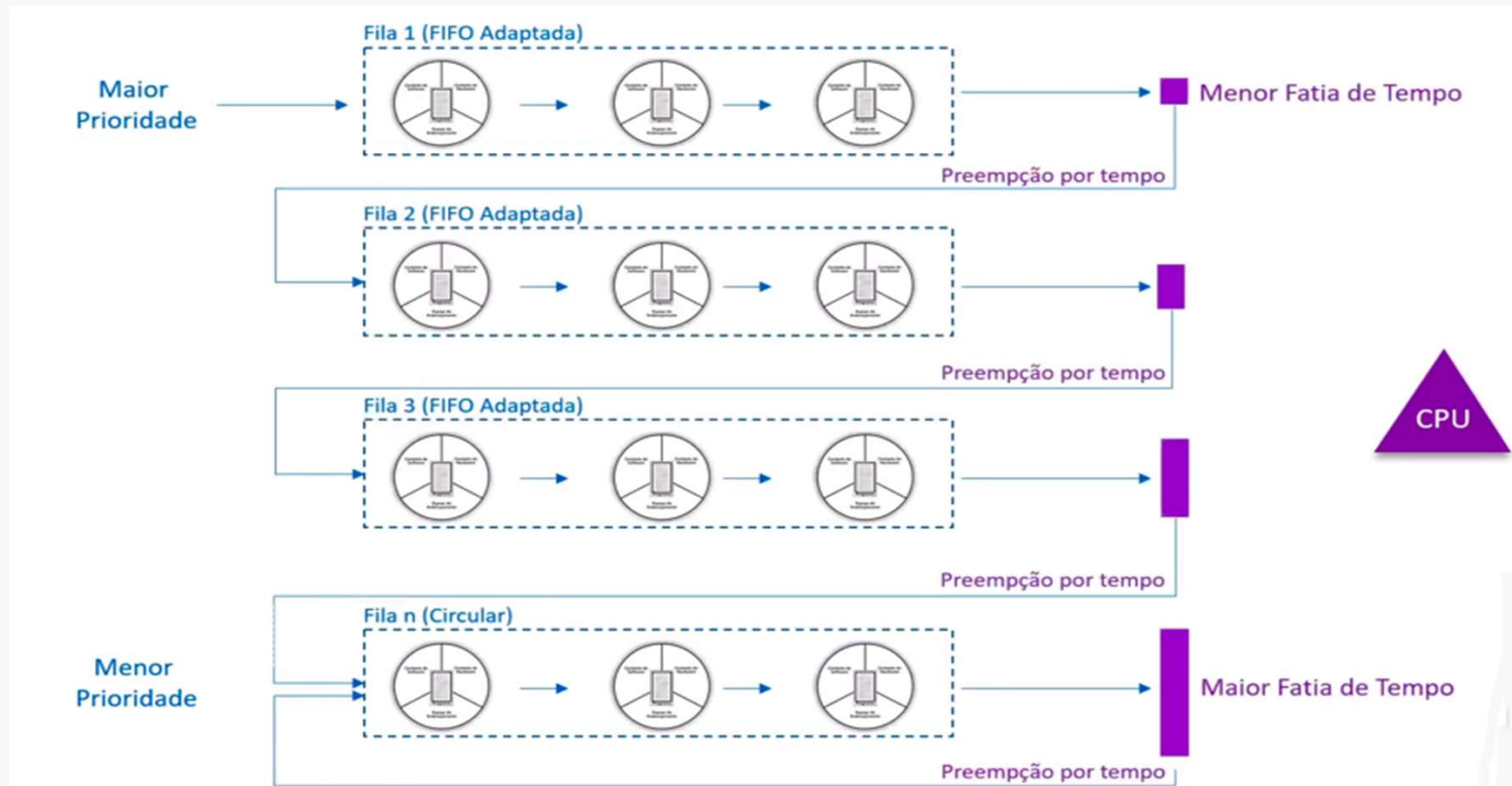


Figura 8: Exemplo de algoritmo de múltiplas filas com realimentação

Algoritmo de Múltiplas Filas

- ❑ Exemplo.: um processo precisa de 100 quantum para ser executado;
 - Inicialmente, ele recebe um quantum para execução;
 - Das próximas vezes ele recebe, respectivamente, 2, 4, 8, 16, 32 e 64 quantum (7 chaveamentos) para execução;
 - Quanto mais próximo de ser finalizado, menos frequente é o processo na CPU.

Algoritmo de Múltiplas Filas –vantagens e desvantagens

- Vantagens:**

- Pode-se implementar políticas sofisticadas;
 - Flexibilidade.

- Desvantagens:**

- Complicado de prever o comportamento;
 - Tem que ter política de promoção e de rebaixamento.

5. Sistemas de Tempo Real

- Sistemas operacionais de tempo real ou RTOS (Real Time Operating Systems) são uma categoria especial de sistemas operacionais.
- São voltados para aplicações onde é essencial a confiabilidade e a execução de tarefas em prazos compatíveis com a ocorrência de eventos externos.

5. Sistemas de Tempo Real

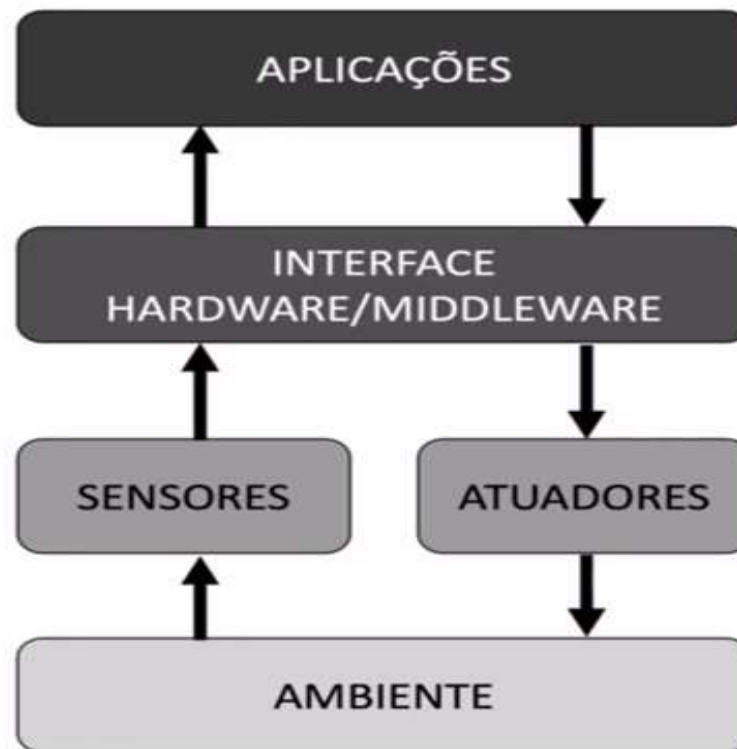


Figura 9: Como funciona os sistemas de tempo real

5. Sistemas de Tempo Real

- É uma classe de sistemas computacionais que exigem corretude lógica e temporal;
- O tempo é um fator crítico;
- Os processos devem cumprir suas computações dentro de um tempo pré-determinado e com rapidez.



Figura 10: Exemplo de sistema de tempo real



Figura 11: Exemplo de sistema de tempo real

Tipos de Sistemas de Tempo Real

- **Soft Real Time Systems (Não críticos):**

- Quando as consequências de uma falha temporal não acarreta em um grande malefício para o funcionamento do sistema.
- Exemplo: Bancos; Multimídia.

- **Hard Real Time Systems (Críticos):**

- Quando as consequências de uma falha temporal excedem os benefícios normais do sistema (falhas catastróficas).
- Exemplo: Aviões, usinas nucleares, hospitais.

Tipos de Sistemas de Tempo Real

- Os processos de tempo real podem ser:
- Periódicos: quando as ativações do processo ocorrem em intervalo regular, chamado de período. As ativações de um processo periódico formam o conjunto de diferentes instâncias do processo.
- Aperiódicos: quando a ativação de um processo responde a eventos internos ou externos definindo uma característica aleatória nessas ativações

Tipos de Sistemas de Tempo Real

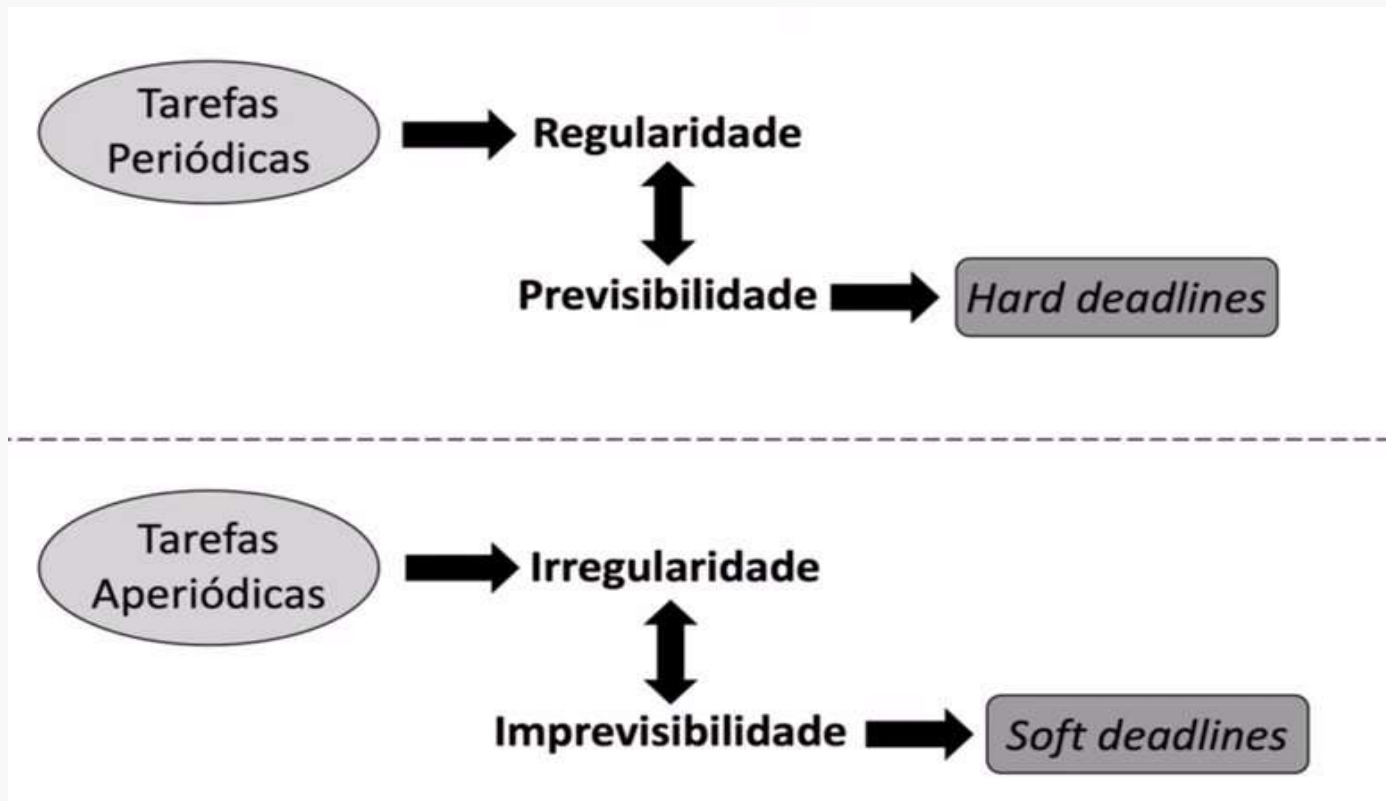


Figura 12: Tipos de sistemas de Tempo Real

Escalonamento em Sistemas de Tempo Real

- Os algoritmos de escalonamento podem ser:
 - Preemptivos ou não-preemptivos;
 - Estáticos: a prioridade de processos é fixa e conhecida antes da execução de processos;
 - Dinâmicos: a prioridade é dinâmica e pode ser modificada ao longo da execução do sistema;
 - Offline: a tomada de decisão de qual processo será executado pode ser feita antes da execução do sistema.
 - Online: a tomada de decisão é em tempo de execução;
- O escalonamento é dividido em duas etapas:
 - Teste de escalonabilidade: determinam se um conjunto de processos é escalonável, ou seja, se o tempo de um conjunto de processos pode ser atendido.
 - Escalonamento: é a própria execução dos processos.

Escalonamento em Sistemas de Tempo Real

- Conceitos básicos:

- Tempo de computação (C_i):

- Tempo necessário para a execução completa da tarefa/processo.

- Tempo de início (S_i):

- Instante de início do processamento da tarefa/processo em uma ativação




- Período (P_i):

- Intervalo de tempo entre as ativações sucessivas de um processo periódico.

- Deadline (D_i):

- Instante máximo desejado para a conclusão de um processo.

Escalonamento em Sistemas de Tempo Real

Tarefas Periódicas	Período/Deadline ($P_i = D_i$)	Tempo de Computação (C_i)	Prioridade RM (p_i)
 Tarefa A	100	20	"Alta"
 Tarefa B	150	40	"Média"
 Tarefa C	350	100	"Baixa"

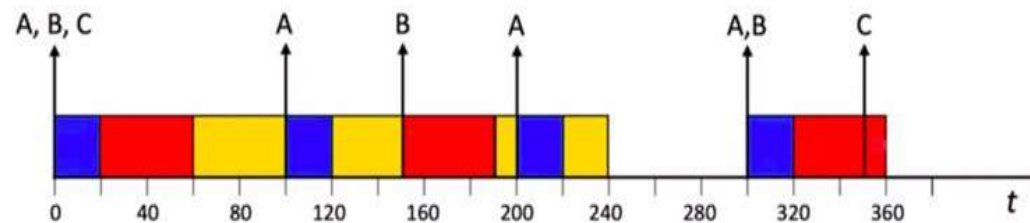


Figura 13: Exemplo de escalonamento em sistemas de tempo real

•Rate Monotonic (RM):

- Preemptivo, estático e online;
- Quanto menor o período, maior a prioridade.
- $D_i = P_i$
- C_i constante

Escalonamento em Sistemas de Tempo Real

- **Algumas desvantagens do Rate Monotonic são:**
 - Não é adequado para tarefas com metas temporais inferiores aos seus períodos, pois pode causar desperdício de recursos ou violação de deadlines.
 - Não considera a importância relativa das tarefas, podendo prejudicar o desempenho de tarefas menos frequentes mas mais críticas.
 - Não se adapta facilmente a variações na carga de trabalho ou na disponibilidade de recursos, pois as prioridades são definidas estaticamente

Escalonamento em Sistemas de Tempo Real

- **Deadline Monotonic (DM):**

- Preemptivo e dinâmico;
- Quanto menor o deadline relativo, maior a prioridade.

- **Earliest Deadline First (EDF):**

- Preemptivo e dinâmico;
- Quanto menor o deadline absoluto, maior a prioridade.

Referências

- TANENBAUM, A. S.; BOS, H. Sistemas Operacionais Modernos. 4ª Edição. Editora Pearson, 2016.
- Priority Scheduling Algorithm. PrepInsta. Disponível em: <https://prepinsta.com/operating-systems/round-robin-scheduling-algorithm/>. Acesso em: 25/10/2023;
- Round Robin Scheduling Algorithm in Operating System. PrepInsta, 2018. Disponível em: <https://prepinsta.com/operating-systems/priority-scheduling-algorithm/>. Acesso em: 25/10/2023;
- Types of Operating System, Tutorials Point. Disponível em: https://www.tutorialspoint.com/operating_system/os_types.htm. Acesso em: 25/10/2023;
- Types of Operating Systems, Geeks for geeks, 2023, Disponível em: <https://www.geeksforgeeks.org/types-of-operating-systems/>. Acesso em: 25/10/2023;
- Farines, Jean-Marie. Sistemas de Tempo Real. Cin UFP, 200. Disponível em: https://www.cin.ufpe.br/~if728/sistemas_tempo_real/livro_farines/cap2.pdf. Acesso em: 25/10/2023.

Read me

Reconhecimentos e Direitos Autorais

@autor: Amanda Mikely Abreu Macedo, Emanuelle da Silva Laune, Gabriela Torres de Queiroz

@data última versão: 09/12/2023

@versão: 1.0

@outros repositórios: <https://github.com/gabrielaqueirxz>

@Agradecimentos: Universidade Federal do Maranhão (UFMA), Professor Doutor Thales Levi Azevedo Valente, e colegas de curso.

@Copyright/License

Este material é resultado de um trabalho acadêmico para a disciplina SISTEMAS OPERACIONAIS, sobre a orientação do professor Dr. THALES LEVI AZEVEDO VALENTE, semestre letivo 2023.2, curso Engenharia da Computação, na Universidade Federal do Maranhão (UFMA). Todo o material sob esta licença é software livre: pode ser usado para fins acadêmicos e comerciais sem nenhum custo. Não há papelada, nem royalties, nem restrições de "copyleft" do tipo GNU. Ele é licenciado sob os termos da licença MIT reproduzida abaixo e, portanto, é compatível com GPL e também se qualifica como software de código aberto. É de domínio público. Os detalhes legais estão abaixo. O espírito desta licença é que você é livre para usar este material para qualquer finalidade, sem nenhum custo. O único requisito é que, se você usá-los, nos dê crédito.

Copyright © 2023 Educational Material

Este material está licenciado sob a Licença MIT. É permitido o uso, cópia, modificação, e distribuição deste material para qualquer fim, desde que acompanhado deste aviso de direitos autorais.

O MATERIAL É FORNECIDO "COMO ESTÁ", SEM GARANTIA DE QUALQUER TIPO, EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS DE COMERCIALIZAÇÃO, ADEQUAÇÃO A UM DETERMINADO FIM E NÃO VIOLAÇÃO. EM HIPÓTESE ALGUMA OS AUTORES OU DETENTORES DE DIREITOS AUTORAIS SERÃO RESPONSÁVEIS POR QUALQUER RECLAMAÇÃO, DANOS OU OUTRA RESPONSABILIDADE, SEJA EM UMA AÇÃO DE CONTRATO, ATO ILÍCITO OU DE OUTRA FORMA, DECORRENTE DE, OU EM CONEXÃO COM O MATERIAL OU O USO OU OUTRAS NEGOCIAÇÕES NO MATERIAL.

Para mais informações sobre a Licença MIT: <https://opensource.org/licenses/MIT>.