

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
BACHARELADO INTERDISCIPLINAR EM CIÊNCIA E TECNOLOGIA
ENGENHARIA DA COMPUTAÇÃO
DISCIPLINA: EECPO010 – SISTEMAS OPERACIONAIS
DOCENTE: PROF. DR. THALES LEVI AZEVEDO VALENTE



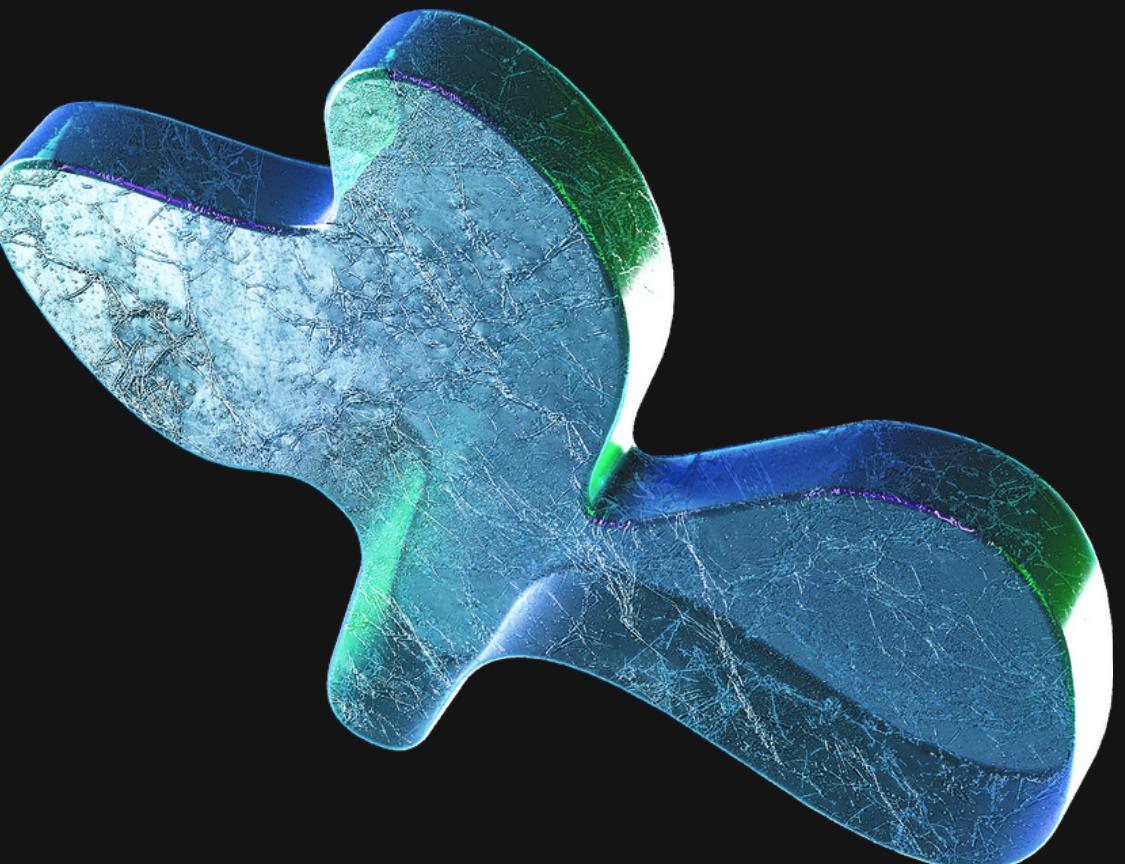
SISTEMAS OPERACIONAIS GERENCIAMENTO DE PROCESSOS

Discentes:

João Pedro Azevedo Veras

Luis Guilherme Freitas de Almeida Silva

Renata Costa Rocha Carvalhal Cavalcante



SUMÁRIO



CONCEITOS BÁSICOS

CATEGORIAS DE
ESCALONAMENTO

SISTEMAS BATCH

SISTEMAS INTERATIVOS

SISTEMAS DE TEMPO REAL

REFERÊNCIAS



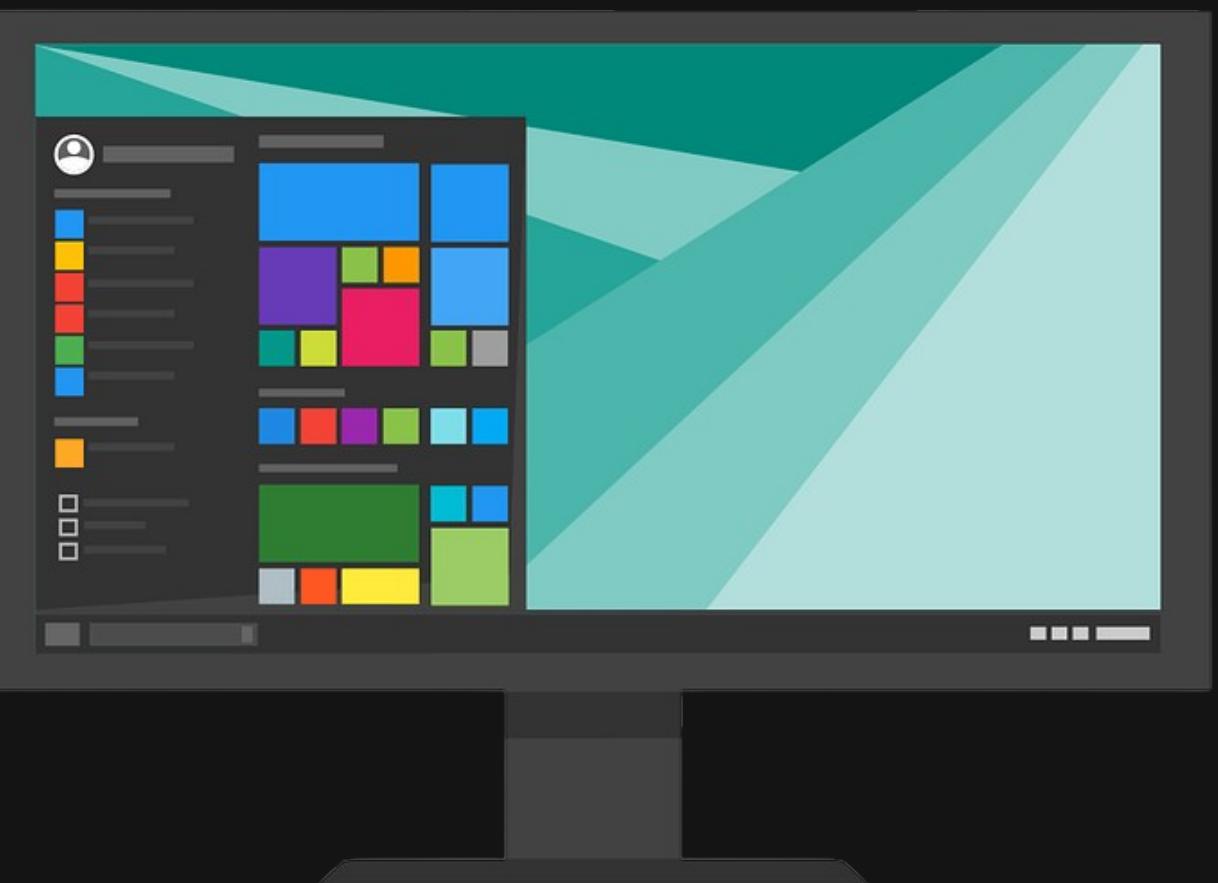


CONCEITOS BÁSICOS

- O QUE É UM SISTEMA OPERACIONAL?

CONCEITOS BÁSICOS

- Programa ou conjunto de programas cuja função é gerenciar os recursos do sistema.
- Atua como uma camada intermediária entre o hardware e os programas executados nele.
- Gerencia os recursos de hardware e software de um computador.





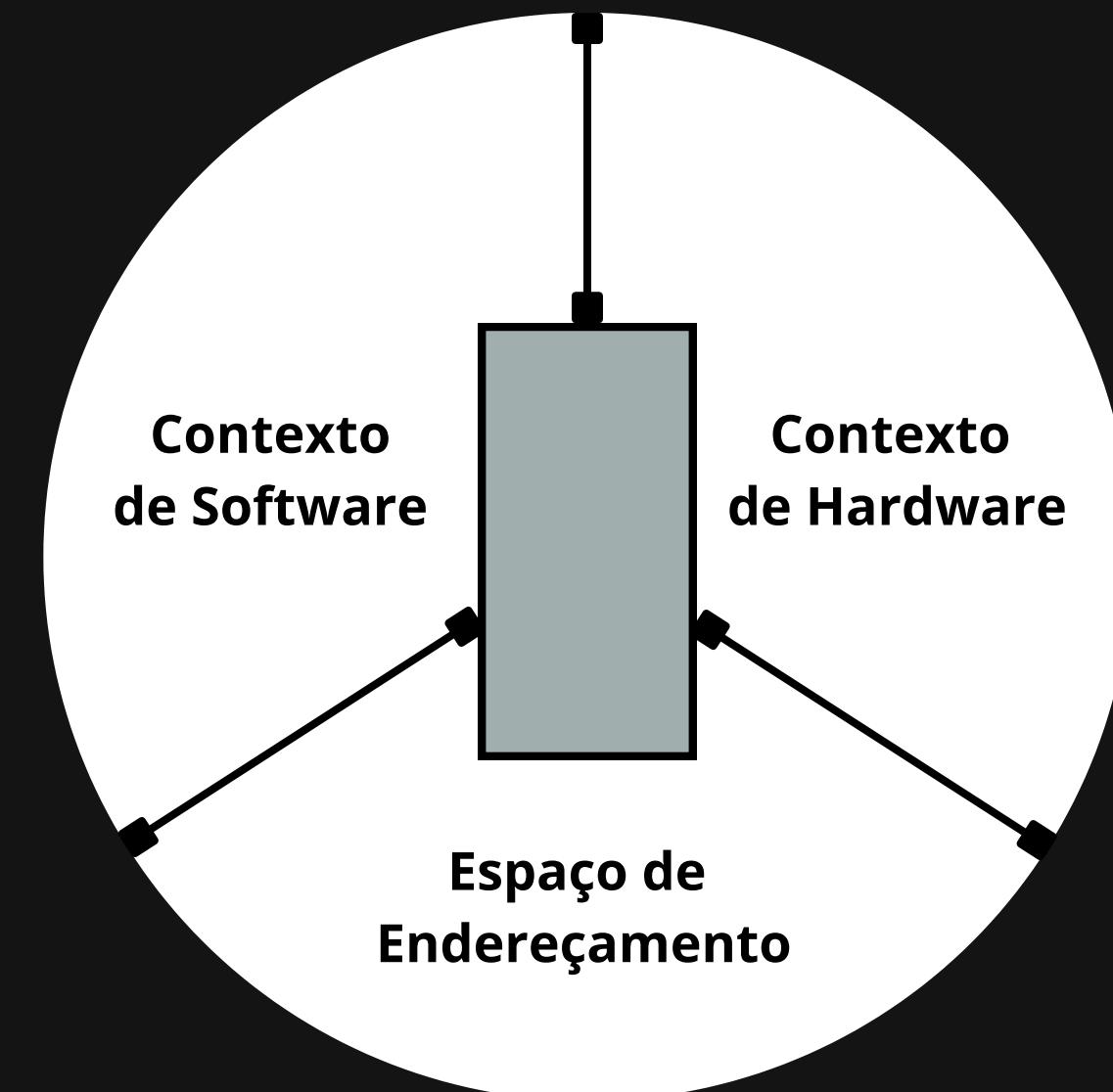
CONCEITOS BÁSICOS

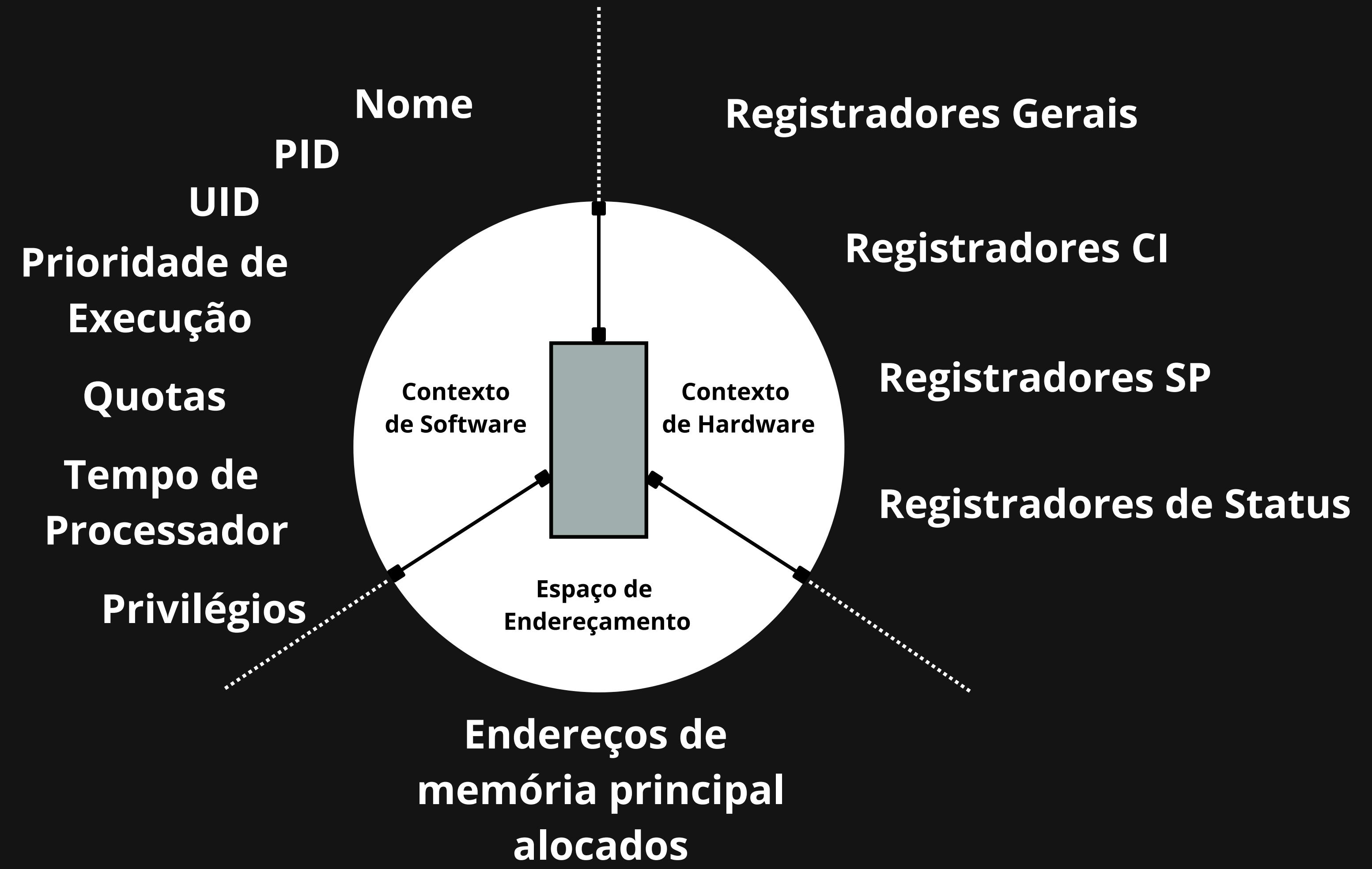
```
ASKS: 243 total, 2 running, 239 sleeping, 2 stopped, 0 zombie
%Cpu(s): 2.8 us, 2.3 sy, 0.0 ni, 91.6 id, 3.3 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 8010248 total, 2563860 free, 2711448 used, 2734940 buff/cache
KiB Swap: 8219644 total, 8211328 free, 8316 used. 4214880 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM TIME+ COMMAND
15121 root      20   0  517420 115160 104260 S  8.6  1.4  2:24.23 Xorg
15718 tautvyd+ 20   0 1173920  98776  58136 S  6.0  1.2  2:30.48 compiz
19505 tautvyd+ 20   0  617384  32004  25148 S  3.0  0.4  0:00.52 gnome-scre
15913 tautvyd+ 20   0 1254200 247236 109348 S  1.0  3.1  5:07.88 chrome
15984 tautvyd+ 20   0 1277680 428300 119400 S  0.7  5.3  2:41.40 chrome
16868 tautvyd+ 20   0 1041548 263828  73944 S  0.7  3.3  1:13.54 chrome
  7 root      20   0      0     0     0 S  0.3  0.0  2:27.84 rcu_sched
15593 tautvyd+ 20   0  633824  35324  25340 S  0.3  0.4  0:03.80 unity-pane
16047 tautvyd+ 20   0  819588  95604  58088 S  0.3  1.2  0:12.78 chrome
16698 tautvyd+ 20   0 1554504 134512  80180 S  0.3  1.7  0:42.05 slack
16815 tautvyd+ 20   0 1057504 251084  96444 S  0.3  3.1  1:33.90 chrome
19497 tautvyd+ 20   0  52200   4132  3416 R  0.3  0.1  0:00.11 top
  1 root      20   0 185424   6072   4000 S  0.0  0.1  0:08.87 systemd
  2 root      20   0      0     0     0 S  0.0  0.0  0:00.19 kthreadd
```

- O QUE É UM PROCESSO?

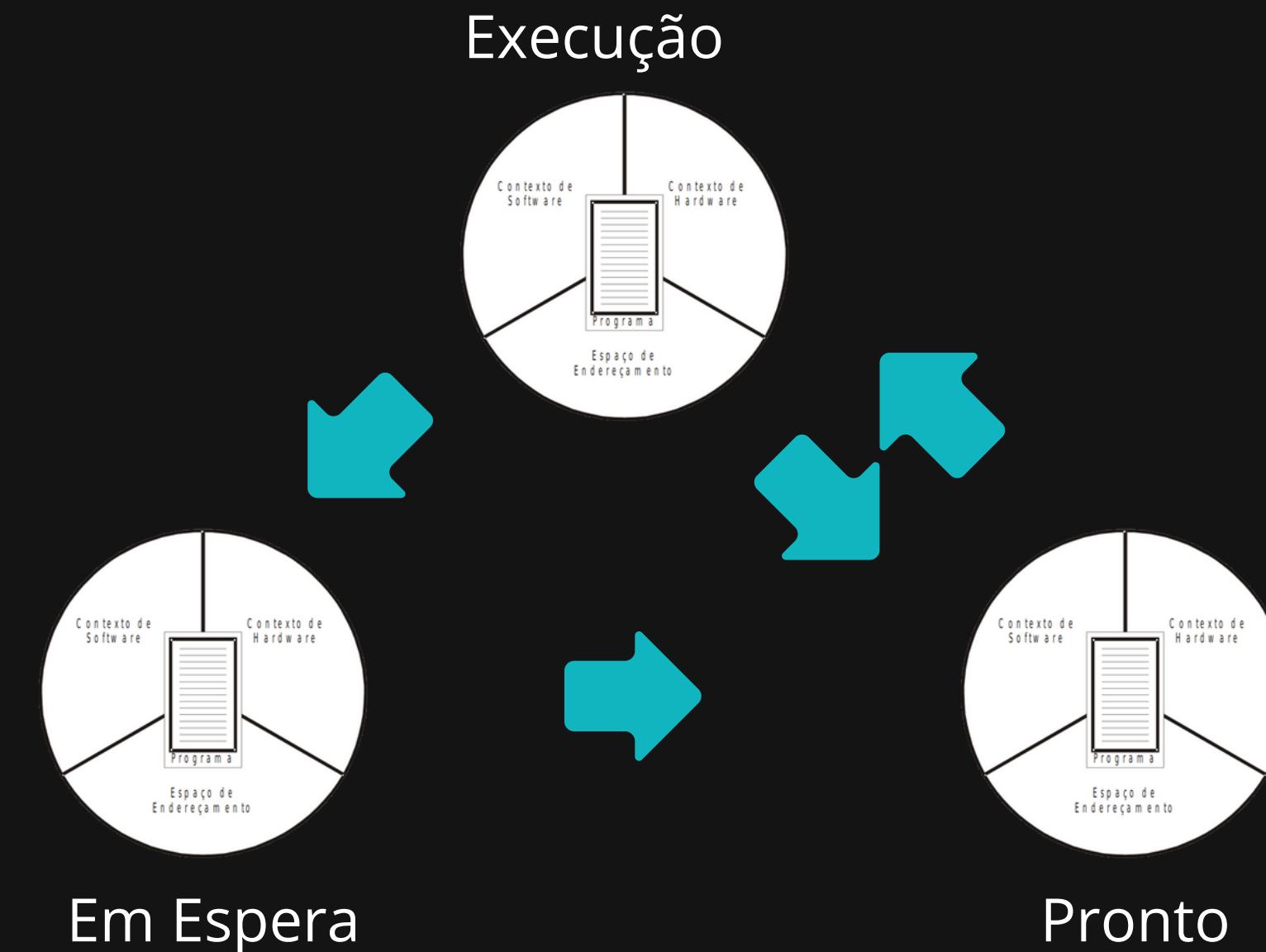
ESTRUTURA DE PROCESSOS





CICLO DE VIDA DE UM PROCESSO

- O ciclo de vida de um processo é uma sequência de estados pelos quais um processo passa desde o momento de sua criação até o término de sua execução. Ele descreve as fases de transição de um processo durante sua vida útil.



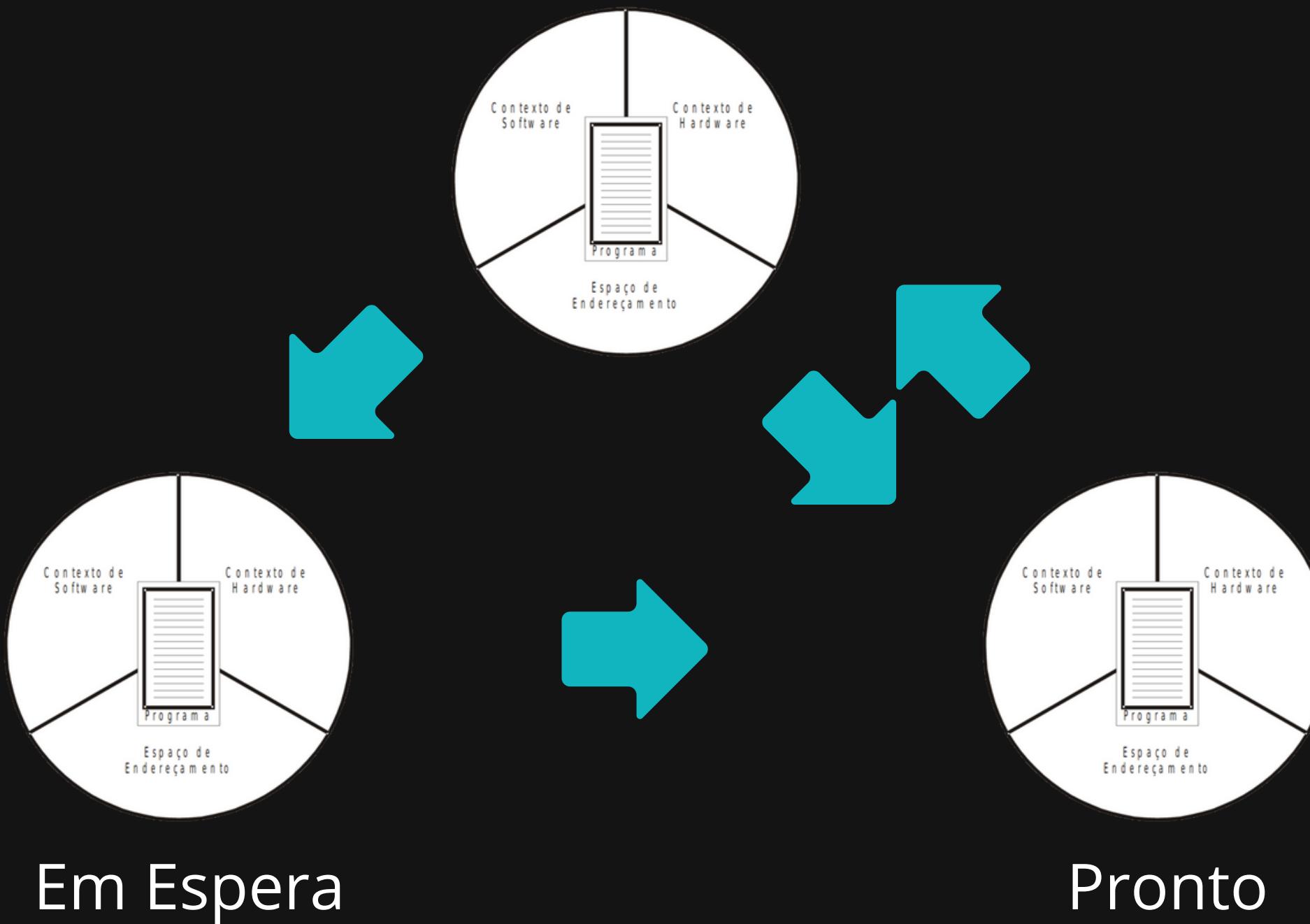
ESTADOS DE UM PROCESSO

- Em Execução
- Pronto
- Em espera

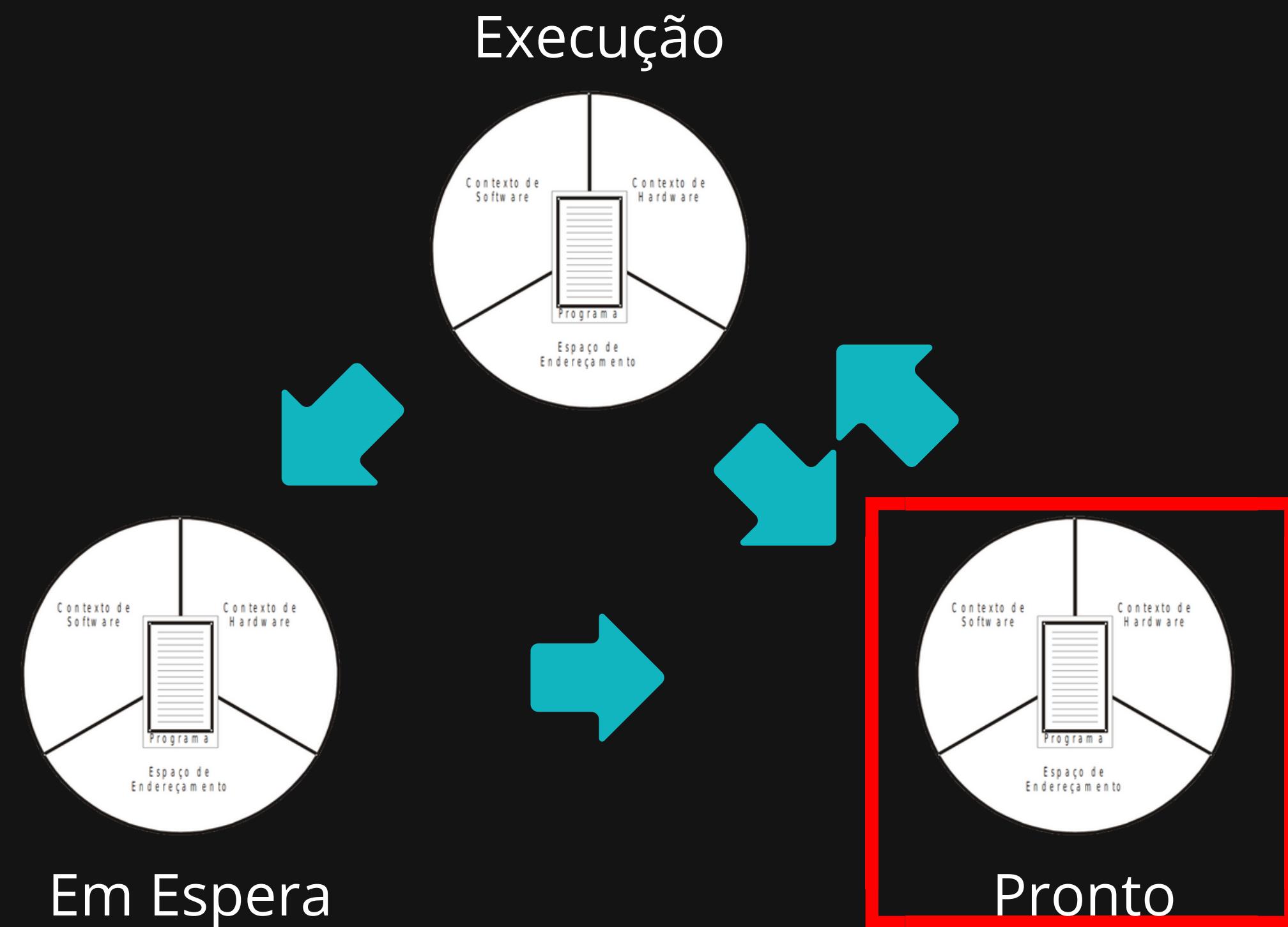


MUDANÇAS DE ESTADO

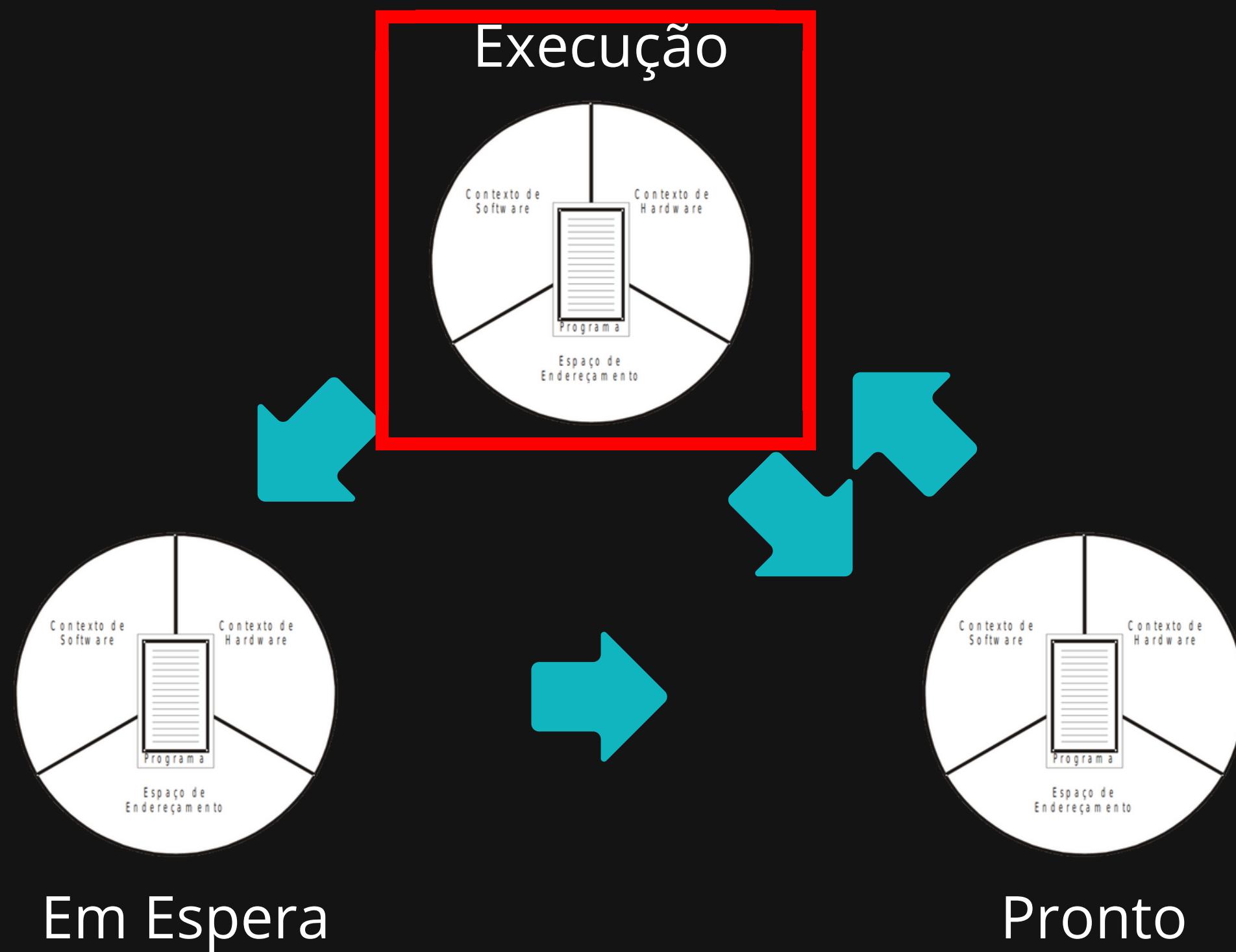
Execução



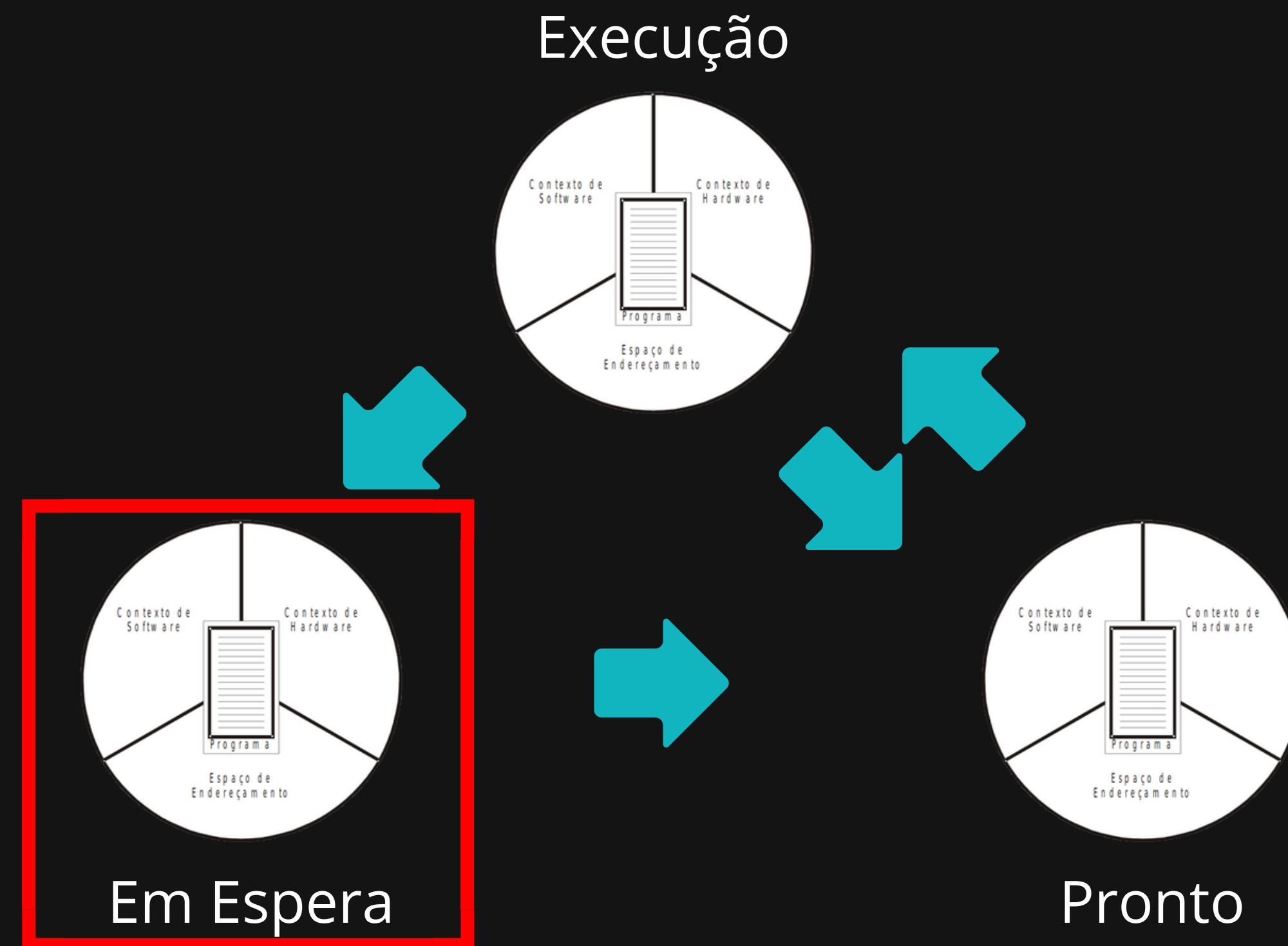
MUDANÇAS DE ESTADO - PRONTO



MUDANÇAS DE ESTADO - EXECUTANDO



MUDANÇAS DE ESTADO - EM ESPERA





CONCEITOS BÁSICOS



- O QUE É ESCALONAMENTO?

1. É UM PROCESSO RESPONSÁVEL POR ESCOLHER OS PROCESSOS QUE SERÃO EXECUTADOS PELO PROCESSADOR

CONCEITOS BÁSICOS



ESCALONADOR

DECIDE O PROCESSO QUE
SERÁ EXECUTADO

DISPATCHER

IMPLEMENTA A DECISÃO DO
ESCALONADOR

MUDANÇA DE CONTEXTO

TROCA DE UM PROCESSO
POR OUTRO

CONCEITOS BÁSICOS

CRITÉRIOS DE
ESCALONAMENTO



UTILIZAÇÃO DO PROCESSADOR

THROUGHPUT

TEMPO DE PROCESSADOR

TEMPO DE ESPERA

TEMPO DE TURNAROUND

TEMPO DE RESPOSTA

CONCEITOS BÁSICOS

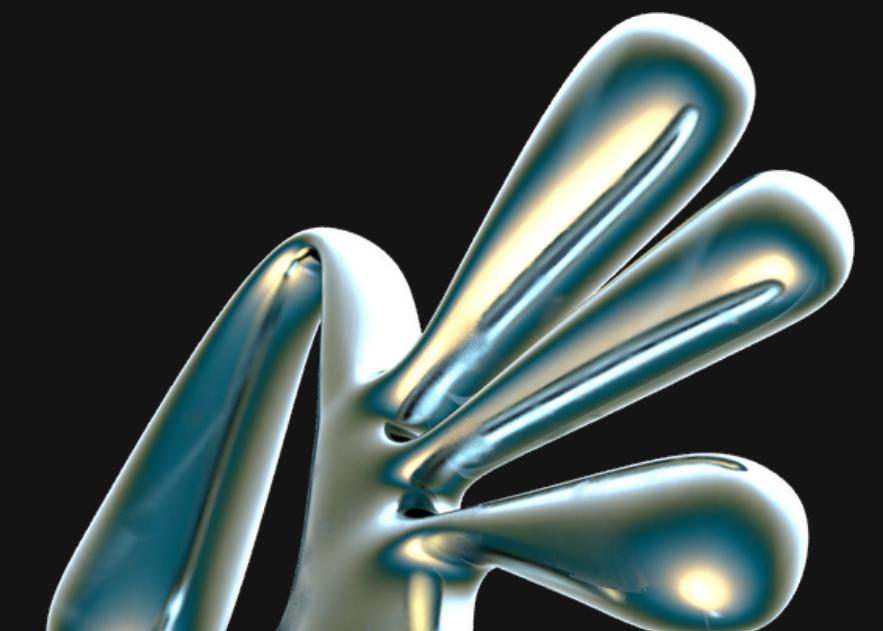
TIPOS DE ESCALONADORES

PREEMPTIVO

- UMA CRIANÇA JOGANDO VÍDEO GAME E SUA MÃE LHE CHAMA PARA ALMOÇAR. SUA GAMEPLAY VAI SER INTERROMPIDA

NÃO-PREEMPTIVO

- UMA CRIANÇA JOGANDO UMA PARTIDA RANQUEADA. A PARTIDA NÃO É INTERROMPIDA



CONCEITOS BÁSICOS

PREEMPTIVO



CONCEITOS BÁSICOS

NÃO-PREEMPTIVO

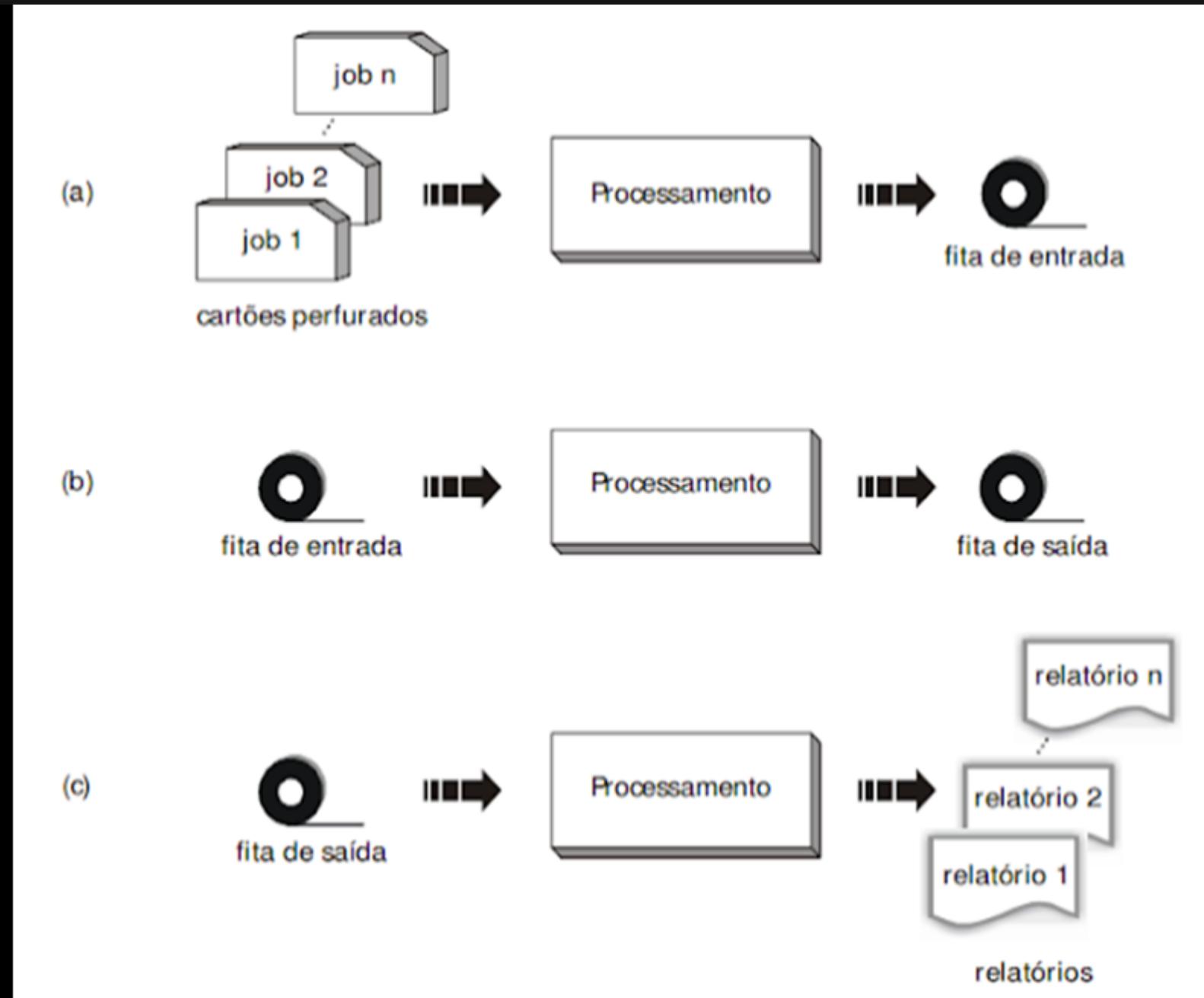


CATEGORIAS DE ESCALONAMENTO



- SISTEMAS BATCH: USUÁRIOS NÃO ESPERAM POR RESPOSTAS RÁPIDAS DO SISTEMA
- SISTEMAS INTERATIVOS: INTERAÇÃO CONSTANTE DOS USUÁRIOS
- SISTEMAS DE TEMPO REAL: PROCESSOS SÃO EXECUTADOS RAPIDAMENTE E O TEMPO É ESSENCIAL

SISTEMAS BATCH



SISTEMAS DE PROCESSAMENTO EM LOTE SÃO SISTEMAS DE COMPUTAÇÃO OU GERENCIAMENTO DE TAREFAS NOS QUAIS UM GRUPO DE TAREFAS É COLETADO, PROCESSADO E EXECUTADO EM LOTE, EM VEZ DE SEREM EXECUTADAS INTERATIVAMENTE EM TEMPO REAL.

ALGORITMOS PARA SISTEMAS EM BATCH

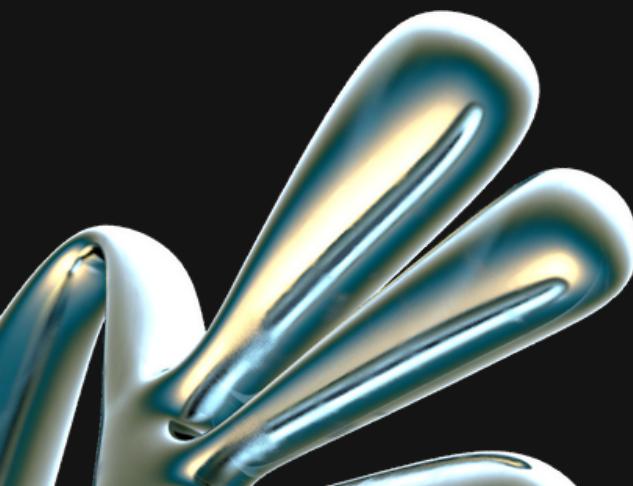
FIRST-COME FIRST-SERVE
(FIFO)



SHORTEST JOB FIRST
(SJF)



SHORTEST REMAINING TIME NEXT(SRTN)



FIRST-IN FIRST-OUT (FIFO)

- NÃO-PREEMPTIVO
- O PROCESSO QUE CHEGAR PRIMEIRO AO ESTADO DE PRONTO É O SELECIONADO PARA A EXECUÇÃO
- PROCESSOS NOVOS VÃO PARA O FINAL DA FILA
- ALGORITMO DE FILA

FIRST-IN FIRST-OUT (FIFO)

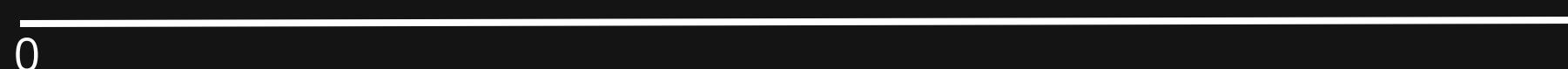
PROCESSO	TEMPO DE CPU
A	9

0



FIRST-IN FIRST-OUT (FIFO)

PROCESSO	TEMPO DE CPU
A	9
B	4



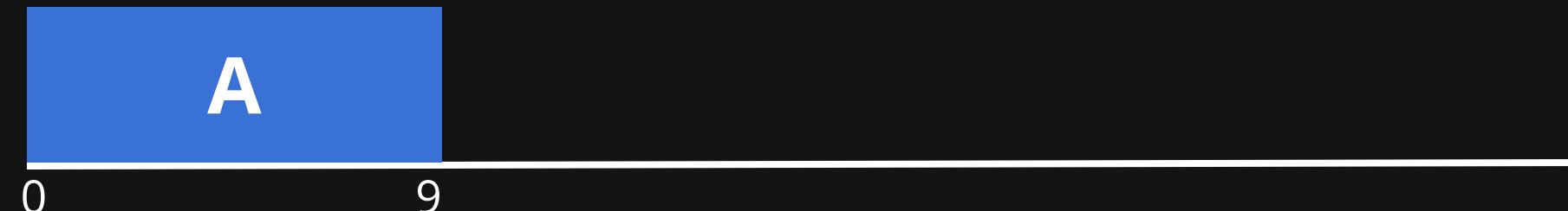
FIRST-IN FIRST-OUT (FIFO)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



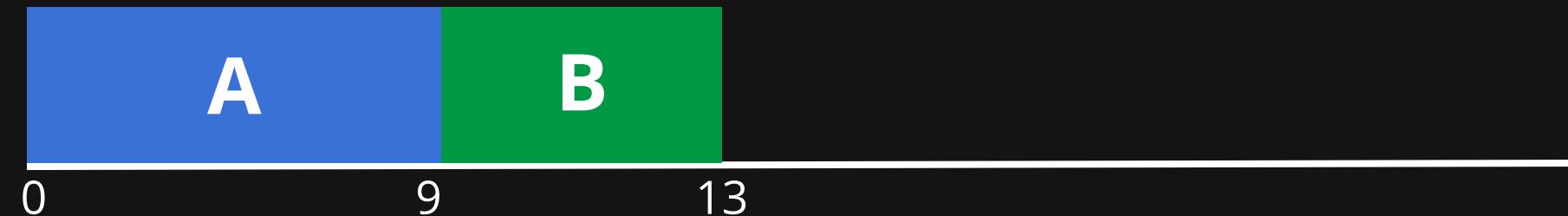
FIRST-IN FIRST-OUT (FIFO)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



FIRST-IN FIRST-OUT (FIFO)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



FIRST-IN FIRST-OUT (FIFO)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



SHORTEST JOB FIRST (SJF)

- Não-preemptivo
- Deve-se prever o tempo de execução de cada processo
- Menor processo é executado primeiro
- Menor Turnaround
- Se possui poucos processos para serem executados o sistema apresenta baixo aproveitamento

SHORTEST JOB FIRST (SJF)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



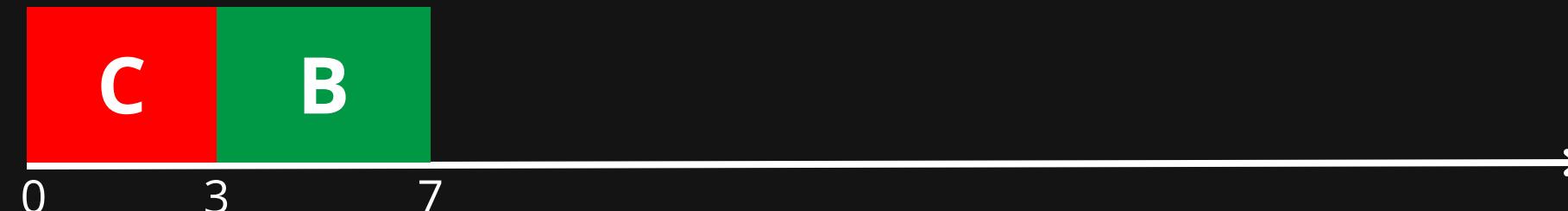
SHORTEST JOB FIRST (SJF)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



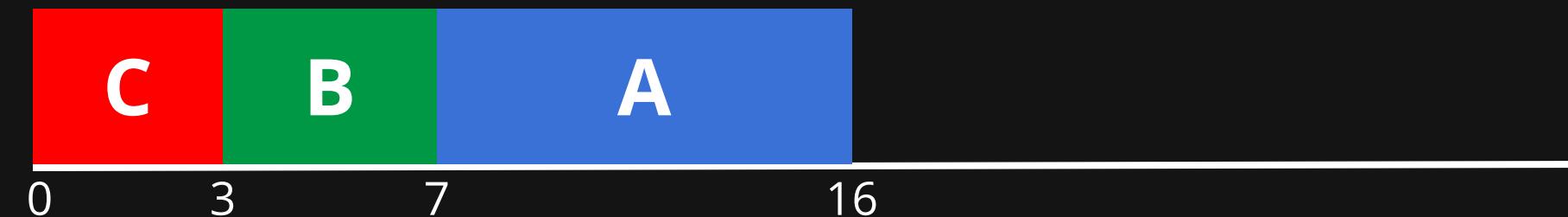
SHORTEST JOB FIRST (SJF)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



SHORTEST JOB FIRST (SJF)

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



SHORTEST REMAINING TIME NEXT (SRTN)

- Preemptivo
- Processos com menor tempo de execução são executados primeiro
- Caso um processo de tempo de execução menor que o processo que atualmente está em execução surja, o processo em execução é interrompido para que o de menor tempo seja executado (toma o controle)

SHORTEST REMAINING TIME NEXT (SRTN)

PROCESSO	TEMPO DE CPU
A	9
B	6



SHORTEST REMAINING TIME NEXT (SRTN)

PROCESSO	TEMPO DE CPU
A	9
B	6
C	3



SHORTEST REMAINING TIME NEXT (SRTN)

PROCESSO	TEMPO DE CPU
A	9
B	6
C	3



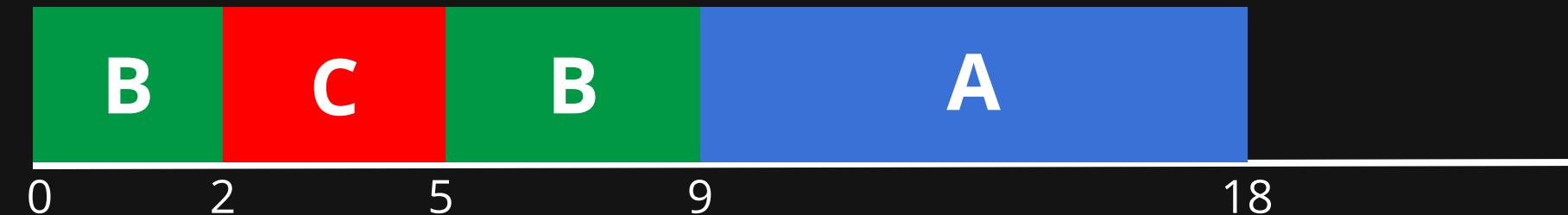
SHORTEST REMAINING TIME NEXT (SRTN)

PROCESSO	TEMPO DE CPU
A	9
B	6
C	3



SHORTEST REMAINING TIME NEXT (SRTN)

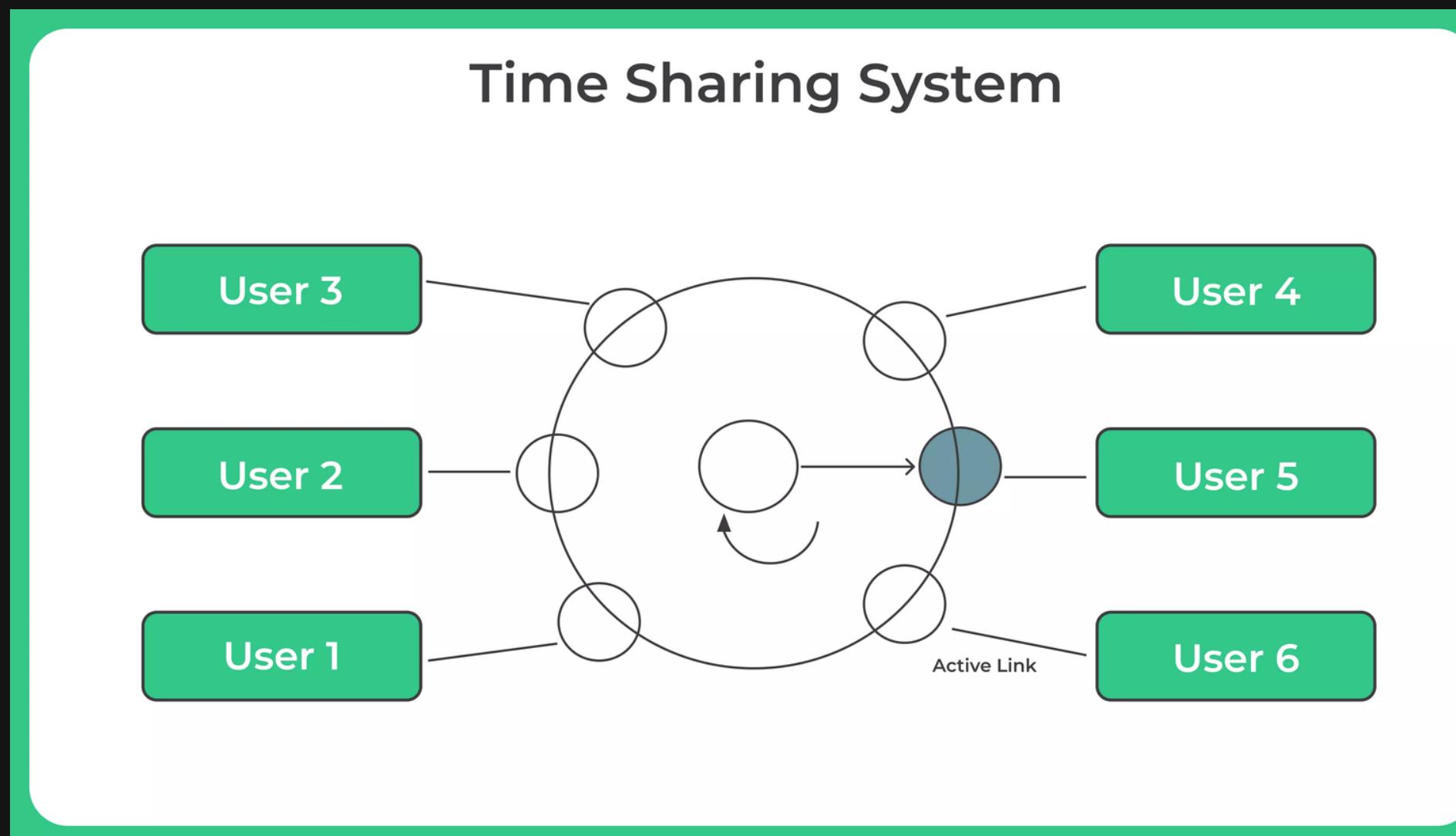
PROCESSO	TEMPO DE CPU
A	9
B	6
C	3



SISTEMAS INTERATIVOS

- VISAM A DIVISÃO DO TEMPO DE PROCESSAMENTO ENTRE DIVERSOS USUÁRIOS
- INTERAÇÃO DIRETA DO USUÁRIO COM O SISTEMA
- SENSAÇÃO DO SISTEMA INTEIRAMENTE DEDICADO AO USUÁRIO

SISTEMAS INTERATIVOS



VANTAGENS

- Permite a execução de diversos programas a partir da divisão de tempo do processador em pequenos intervalos
- Interação do usuário com o sistema enquanto os processos executam

DESVANTAGENS

- Maior sobrecarga em relação a outros S.O, pois necessita de agendamento, alternância de contexto e suporte a vários usuários.

SISTEMAS INTERATIVOS – Round Robin

- CADA PROCESSO RECEBE UMA QUANTIDADE DE TEMPO FIXO PARA USAR A CPU CHAMADO DE QUANTUM
- PREEMPTIVO
- POSSUI BOM TEMPO DE RESPOSTA

SISTEMAS INTERATIVOS - ROUND ROBIN

- Se o Quantum $\rightarrow \infty$ obtém-se o comportamento de um escalonador FIFO
- Quanto menor o Quantum maior o número de trocas de contexto, assim, diminuindo a eficiência da CPU
- Caso o Quantum seja muito grande o tempo de resposta é comprometido

SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	9
B	4
C	3



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	7 ↓
B	4
C	3



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

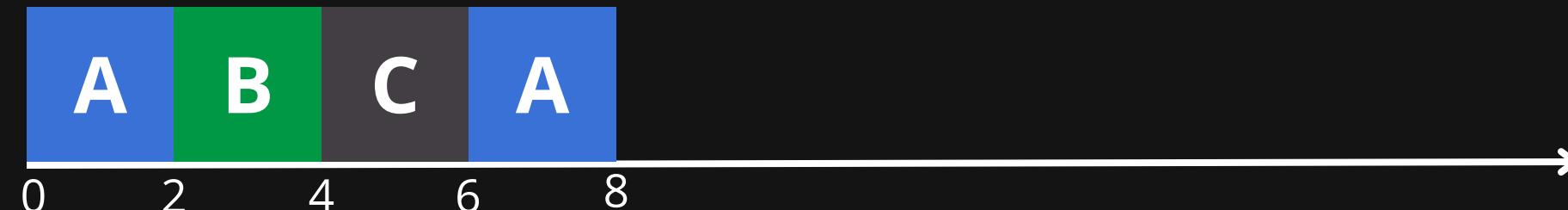
PROCESSO	TEMPO DE CPU
A	7
B	2
C	3



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

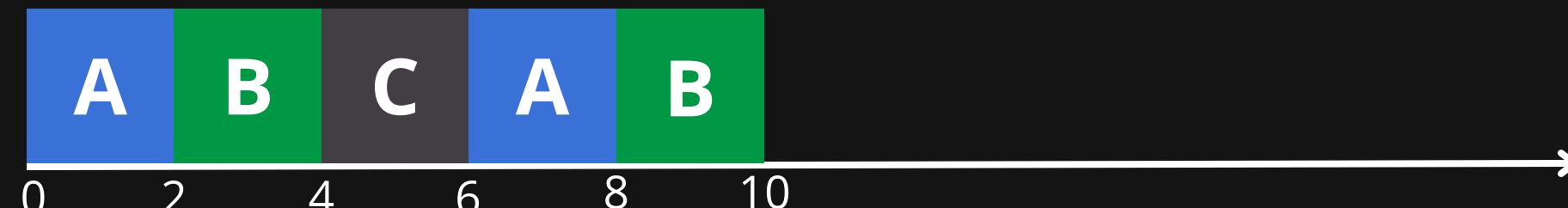
PROCESSO	TEMPO DE CPU
A	7
B	2
C	1



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

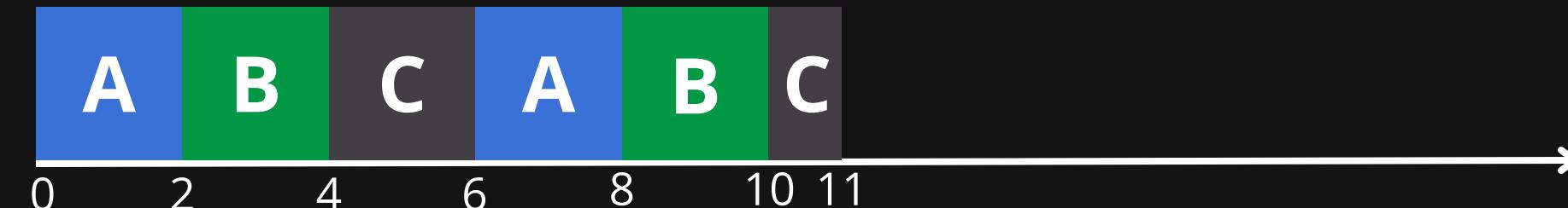
PROCESSO	TEMPO DE CPU
A	5 ↓
B	2
C	1



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	5
B	0
C	1



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	5
B	0
C	0



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

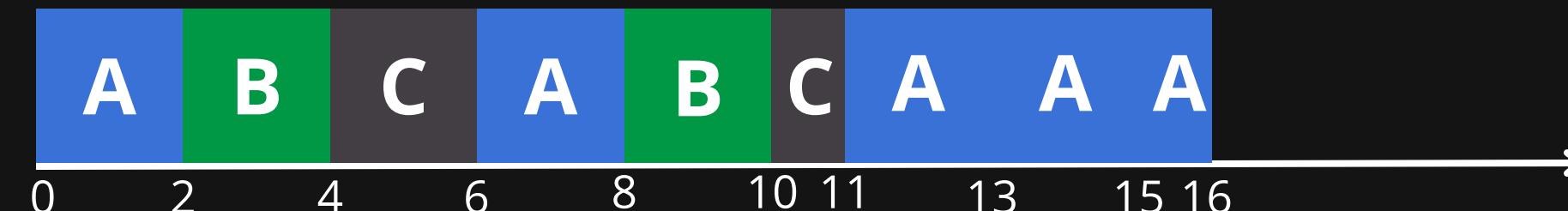
PROCESSO	TEMPO DE CPU
A	3 ↓
B	0
C	0



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	1 ↓
B	0
C	0



SISTEMAS INTERATIVOS - ROUND ROBIN

Exemplo com Quantum = 2 ms

PROCESSO	TEMPO DE CPU
A	0 ↓
B	0
C	0



Tempo máximo de espera = 5 ms

SISTEMAS INTERATIVOS – PRIORIDADE

- Cada processo possui uma prioridade de execução, processos com maior prioridade são executados primeiro
- Quanto menor for o número da prioridade, maior ela será
- A medida que o tempo passa, vai-se aumentando a prioridade do processo (aging)
- Processos com prioridades baixas podem demorar muito a serem executados

SISTEMAS INTERATIVOS - PRIORIDADE

Exemplo:

PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	4
B	8	2
C	10	1



SISTEMAS INTERATIVOS - PRIORIDADE

Exemplo:

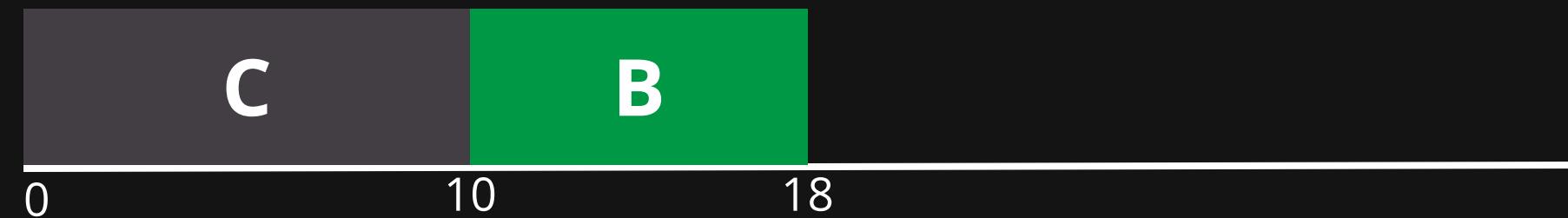
PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	4
B	8	2
C	10	1



SISTEMAS INTERATIVOS - PRIORIDADE

Exemplo:

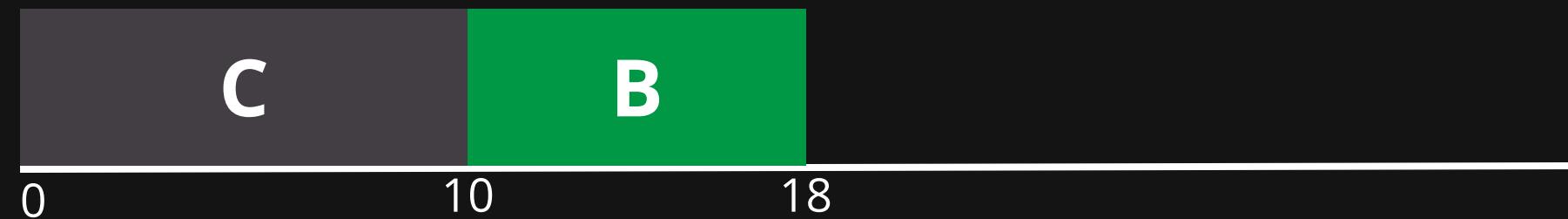
PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	3 ↓
B	8	1 ↓
C	0 ↓	1



SISTEMAS INTERATIVOS - PRIORIDADE

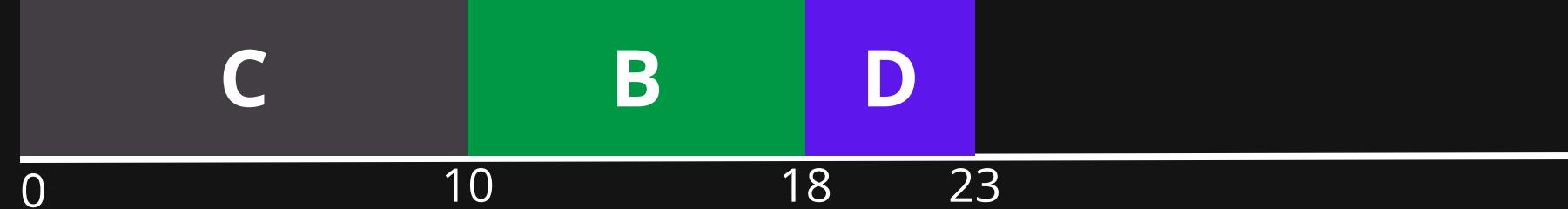
Exemplo:

PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	2 ↓
B	0 ↓	1
C	0	0 ↓



SISTEMAS INTERATIVOS - PRIORIDADE

PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	2
B	0	1
C	0	0
D	5	1



SISTEMAS INTERATIVOS - PRIORIDADE

PROCESSO	TEMPO DE CPU	PRIORIDADE
A	15	1 ↓
B	0	0 ↓
C	0	0
D	0 ↓	1



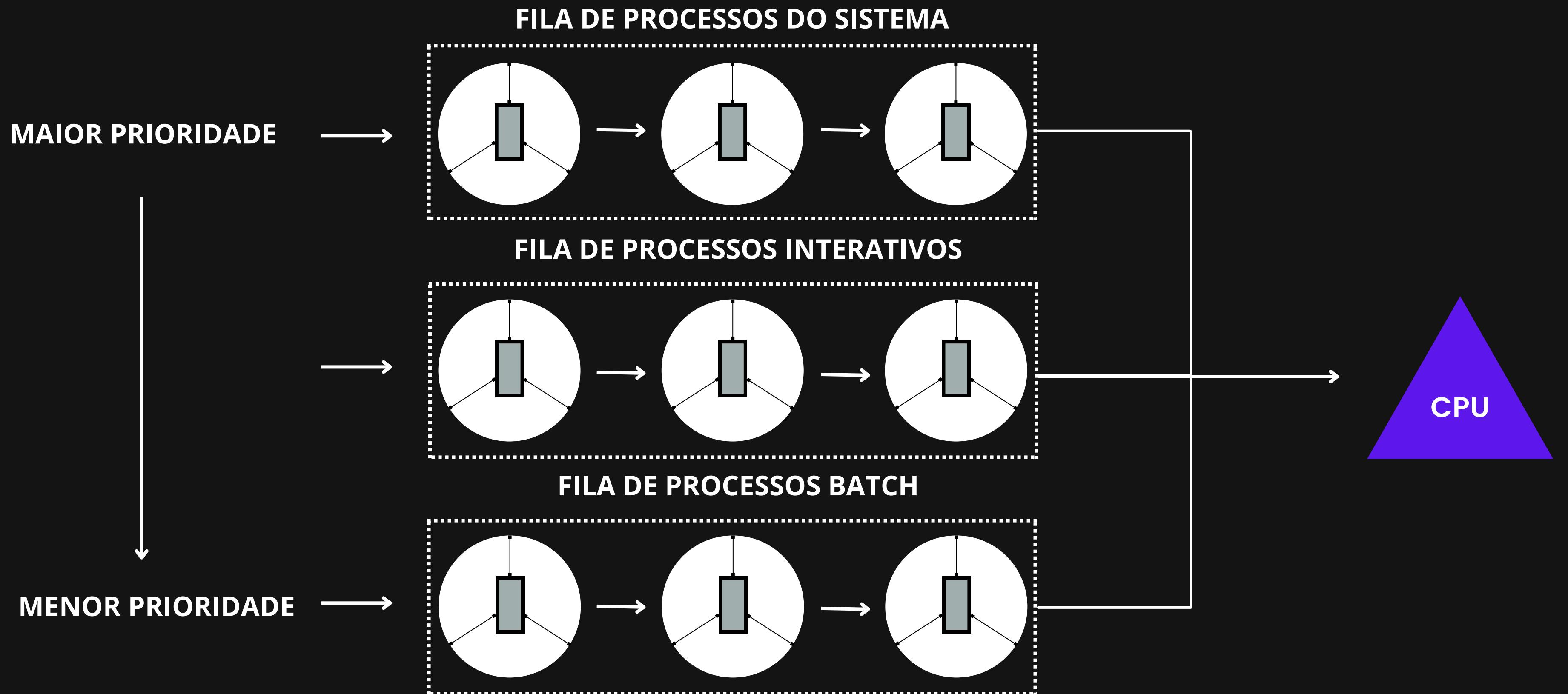
SISTEMAS INTERATIVOS - PRIORIDADE

PROCESSO	TEMPO DE CPU	PRIORIDADE
A	0 	1
B	0	0
C	0	0
D	0	0 



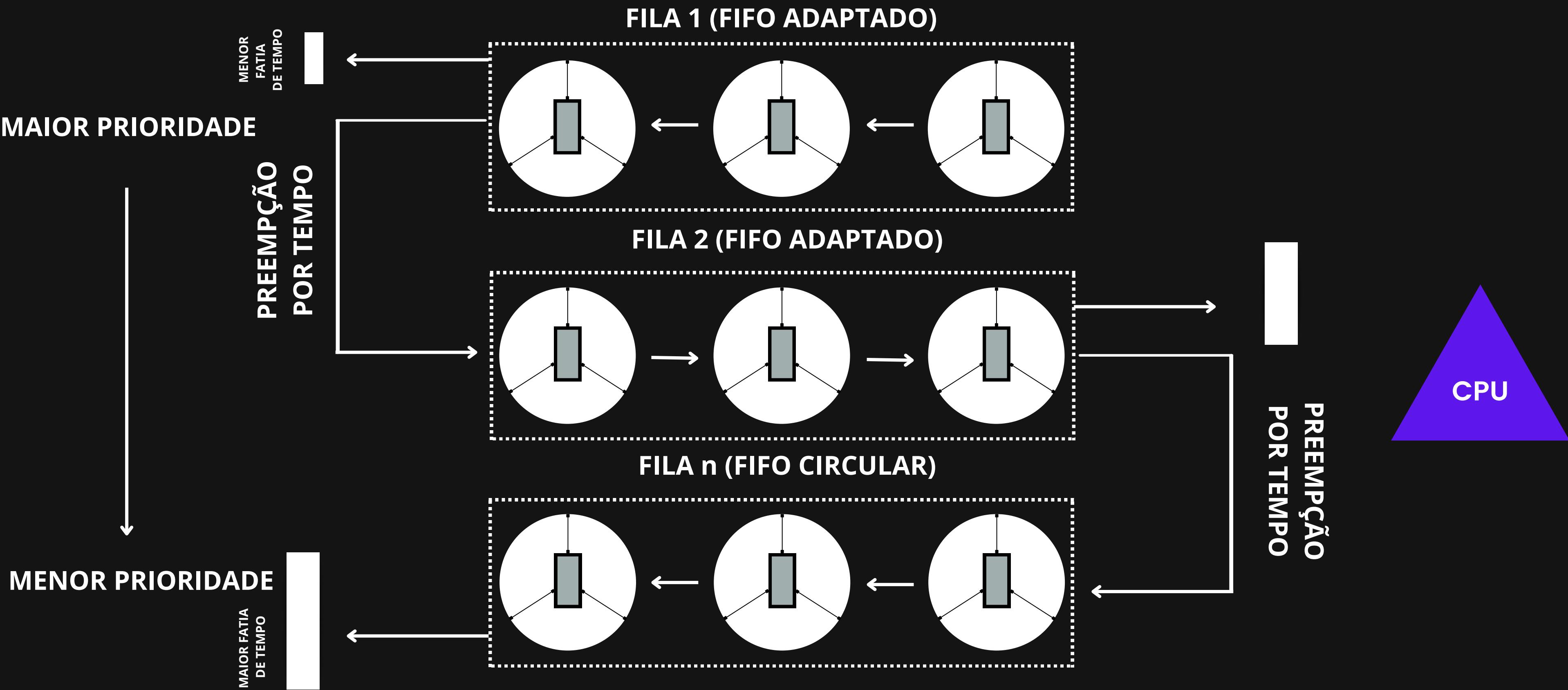
SISTEMAS INTERATIVOS - MULTIPLAS FILAS

- Os processos são divididos em diferentes filas, dependendo da sua prioridade
- O sistema só irá passar para a próxima fila quando a fila de maior prioridade ficar vazia
- Cada fila possui tipos de algoritmos de escalonamentos diferentes

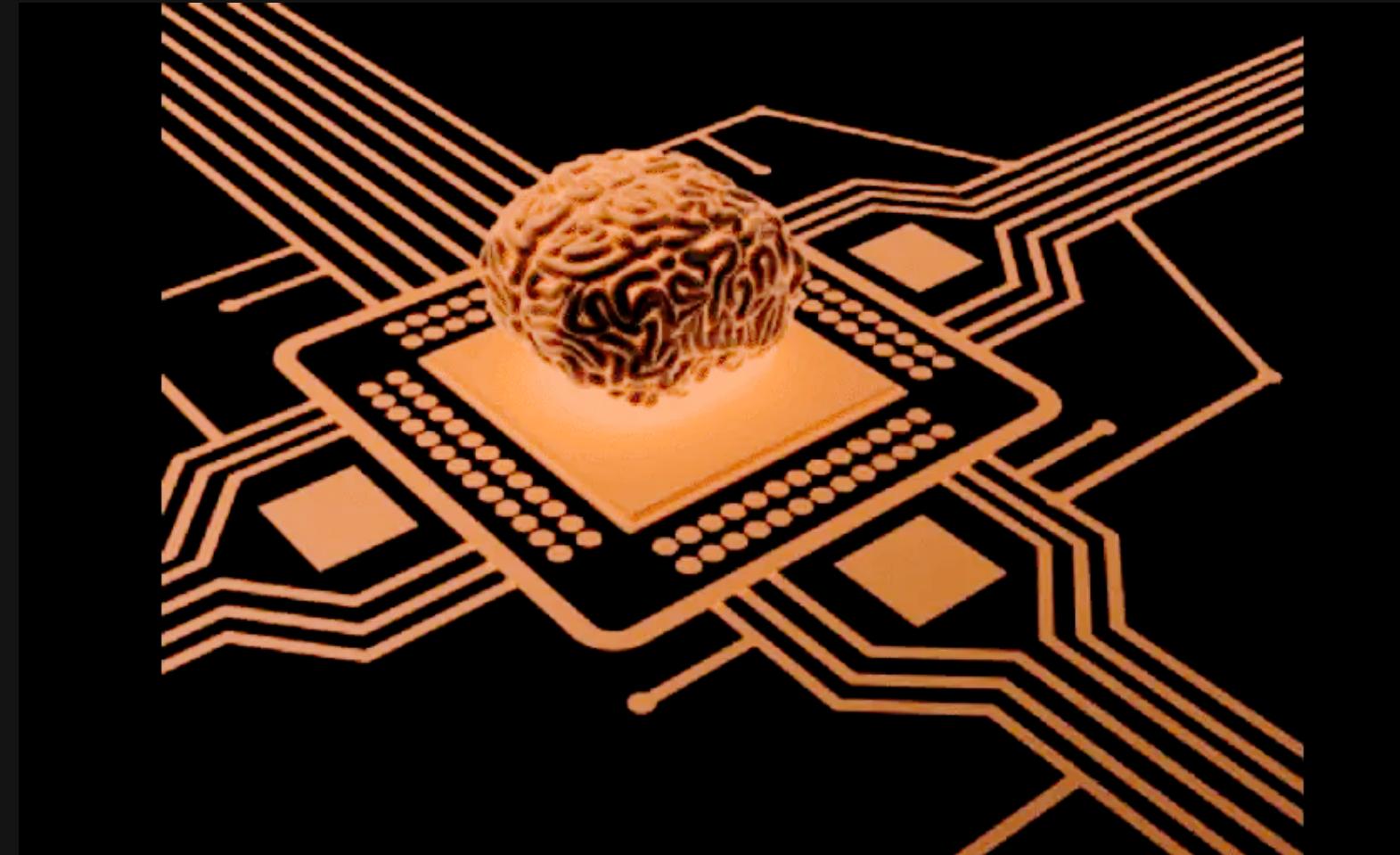


SISTEMAS INTERATIVOS - MULTIPLAS FILAS COM REALIMENTAÇÃO

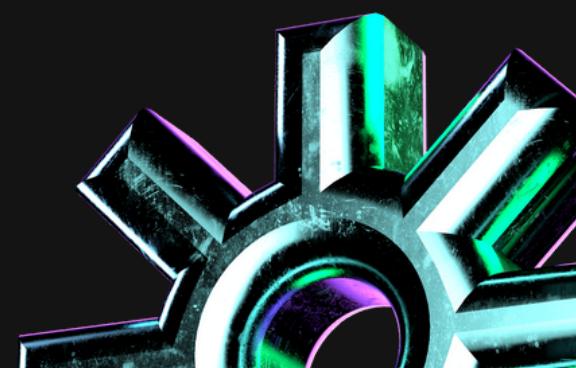
- Também implementa diversas filas onde cada uma tem uma prioridade de execução associada, porém, os processos não permanecem em uma mesma fila até o término do processamento.
- O sistema identifica dinamicamente o comportamento de cada processo, ajustando assim suas prioridades de execução e mecanismos de escalonamento.
- Os processos não são previamente associados às filas de pronto, e sim direcionados pelo sistema entre as diversas filas com base em seu comportamento.



SISTEMAS DE TEMPO REAL



- RTOS (REAL-TIME OPERATING SYSTEM)
É UM SOFTWARE CENTRAL QUE GERENCIA OS RECURSOS DE UM SISTEMA DE COMPUTAÇÃO, GARANTINDO QUE TODAS AS TAREFAS SEJAM CONCLUÍDAS DENTRO DE PRAZOS ESPECÍFICOS E DE MANEIRA EFICIENTE.
- "KERNEL" OU "NÚCLEO" DO SISTEMA DE TEMPO REAL.
- GESTÃO PRECISA E OPORTUNA DE EVENTOS CRÍTICOS EM SISTEMAS DE TEMPO REAL.



SISTEMAS DE TEMPO REAL

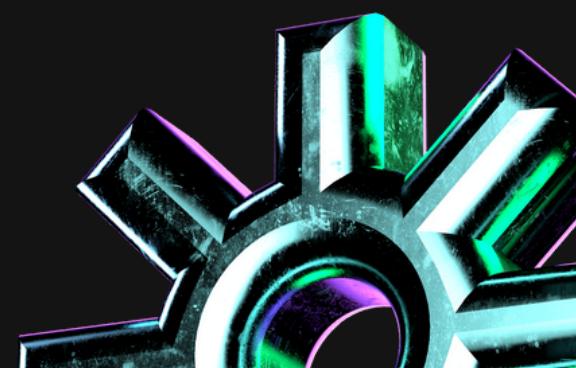


SISTEMAS DE TEMPO REAL

DEVEM OPERAR DE FORMA PREVISÍVEL E ATENDER A RESTRIÇÕES DE TEMPO ESPECÍFICAS. CARACTERÍSTICAS:



CORRETUDE LÓGICA E TEMPORAL;
TEMPO É CRÍTICO;
RAPIDEZ;
DIVISÃO EM TAREFAS;
HARDWARE ESPECÍFICO;
PREEMPTIVIDADE;
USO DE SEMÁFOROS;
USO DE BUFFERS.



SISTEMAS DE TEMPO REAL

- **TAREFA (TASK)**

Segmento de código cuja execução possui atributo temporal próprio; Exemplo: Sub-rotina, trecho de um programa.

- **DEADLINE**

Instante máximo desejado para a conclusão de uma tarefa.

- **TEMPO REAL CRÍTICO (HARD REAL-TIME)**

Falha temporal pode resultar em consequências catastróficas;

Necessário garantir requisitos temporais em projeto;

Exemplo: Usina nuclear, indústria petroquímica, mísseis.

- **TEMPO REAL NÃO CRÍTICO (SOFT REAL-TIME)**

Requisito temporal descreve apenas comportamento desejado;

Exemplo: Multimídia.

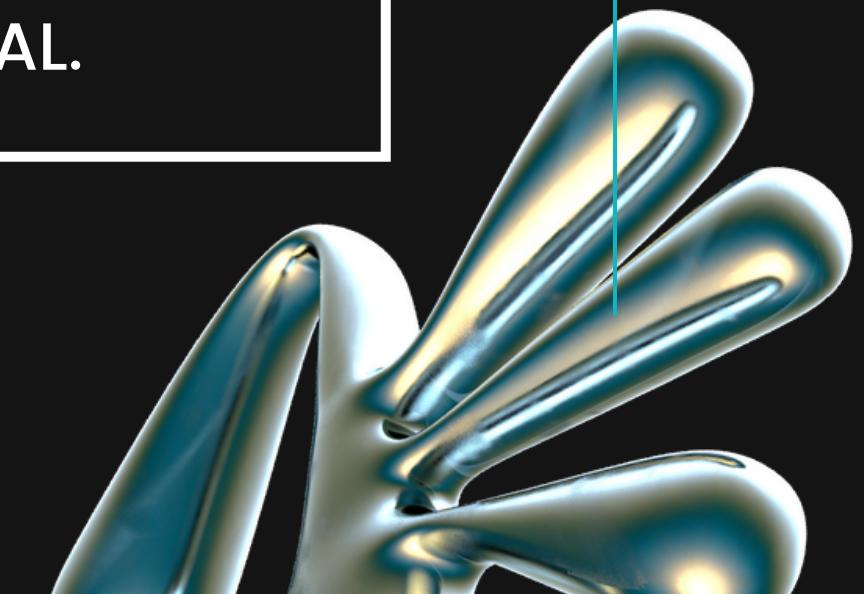


SOFT REAL-TIME

FLEXIBILIDADE TEMPORAL

DESEMPENHO BALANCEADO

VIDEOCONFERÊNCIAS, STREAMING DE VÍDEO, JOGOS ON-LINE E APLICATIVOS DE ENTRETENIMENTO EM GERAL.



HARD REAL-TIME

RESTRIÇÕES TEMPORAIS RÍGIDAS

PRIORIDADE NAS RESTRIÇÕES TEMPORAIS

EXEMPLOS: CONTROLE DE TRÁFEGO AÉREO,
FRENAGEM DE VEÍCULOS AUTOMOTORES,
USINAS NUCLEARES E DISPOSITIVOS
MÉDICOS.

SEGURANÇA CRÍTICA

CLASSIFICAÇÃO PROCESSOS DE SISTEMAS DE TEMPO REAL

PERIÓDICOS

TAREFAS OCORREM EM INTERVALOS REGULARES E PREVISÍVEIS.

CADA TAREFA TEM UM PERÍODO FIXO ENTRE SUAS ATIVAÇÕES.

AS RESTRIÇÕES TEMPORAIS SÃO BEM DEFINIDAS E RÍGIDAS.

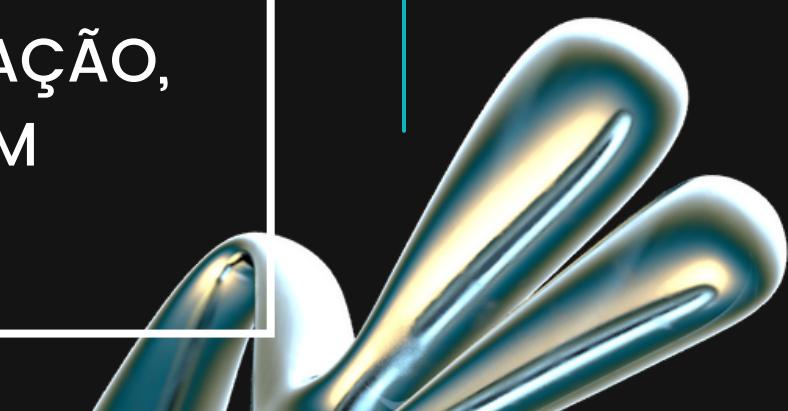
CONTROLE INDUSTRIAL, SISTEMAS DE COLETA DE DADOS AMBIENTAIS E SISTEMAS DE AUTOMAÇÃO, ONDE AS LEITURAS OU AÇÕES OCORREM REGULARMENTE E REPETIDAMENTE.

NÃO-PERIÓDICOS

TAREFAS OU EVENTOS NÃO SEGUEM UM PADRÃO REGULAR.

AS RESTRIÇÕES TEMPORAIS PODEM VARIAR, MAS AINDA SÃO CRÍTICAS.

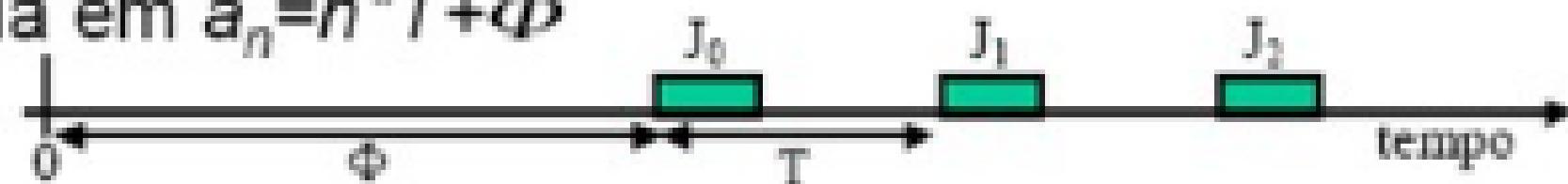
SISTEMAS DE DETECÇÃO DE FALHAS EM EQUIPAMENTOS, SISTEMAS DE SEGURANÇA.



CLASSIFICAÇÃO PROCESSOS DE SISTEMAS DE TEMPO REAL

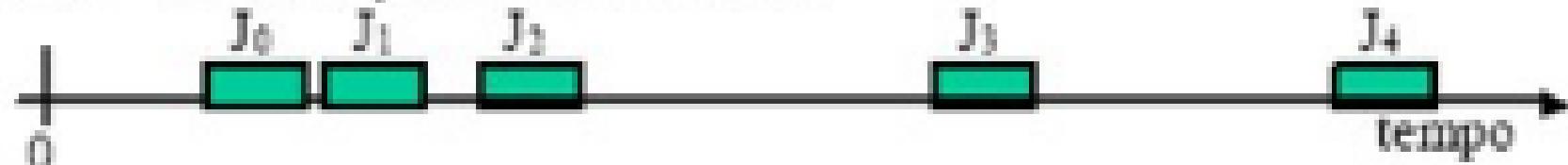
- periódicas

instância n activada em $a_n = n \cdot T + \phi$



- aperiódicas

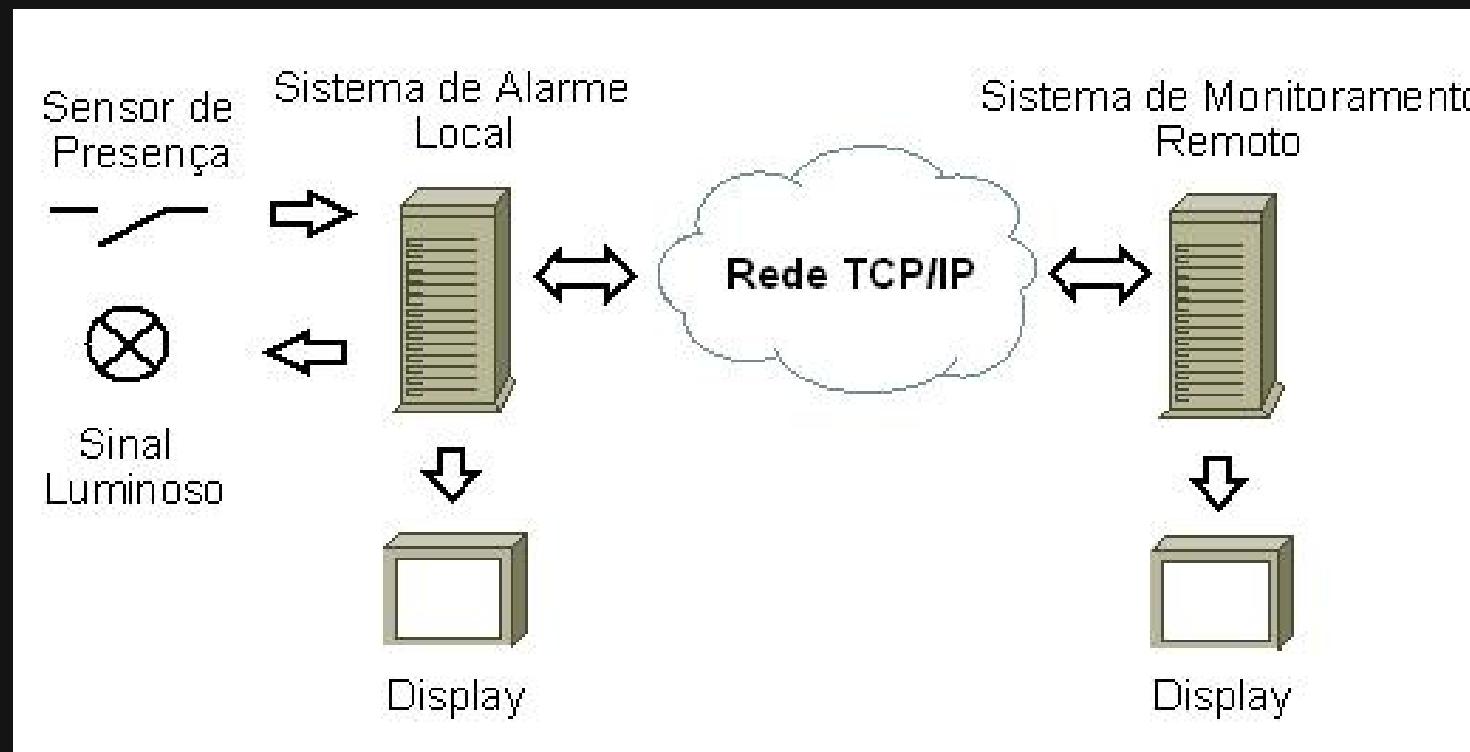
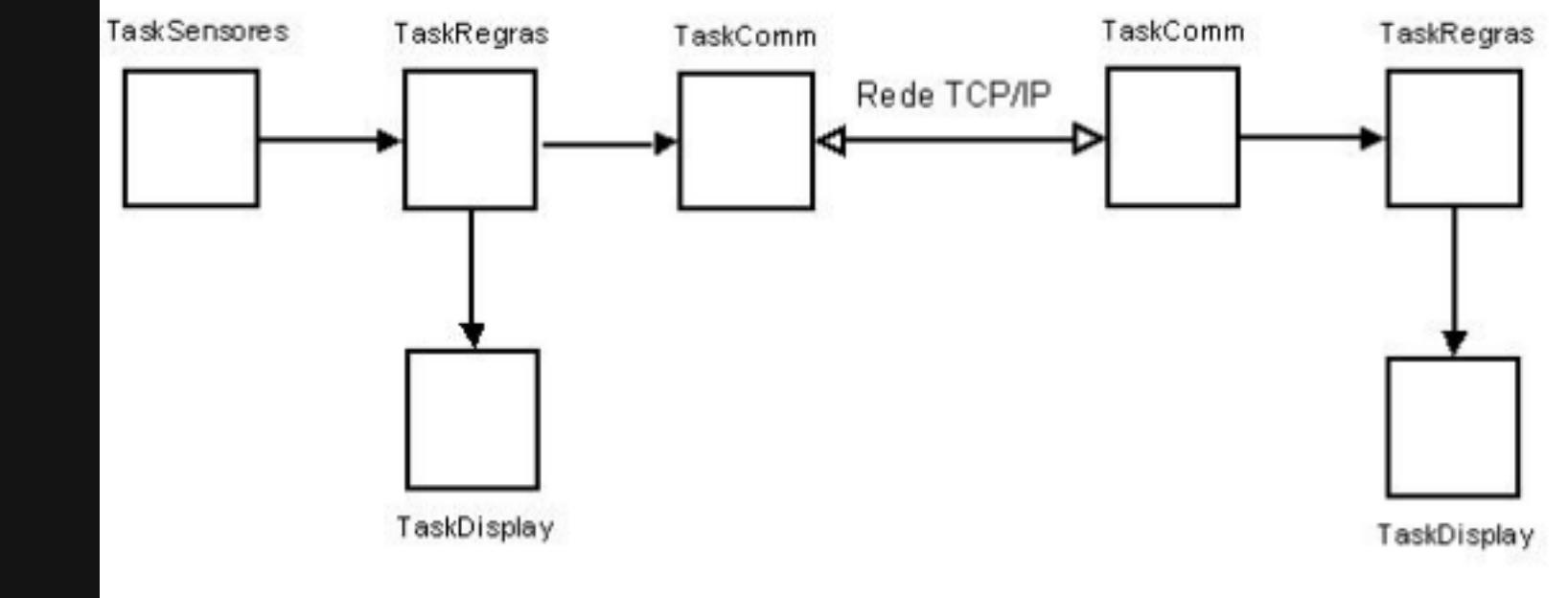
só se caracterizam de forma probabilística



ESCALONAMENTO EM SISTEMAS DE TEMPO REAL



TÉCNICA DE ATRIBUIR PRIORIDADES E DETERMINAR A ORDEM DE EXECUÇÃO DAS TAREFAS OU PROCESSOS EM UM SISTEMA QUE OPERA SOB RESTRIÇÕES DE TEMPO. GARANTE QUE AS TAREFAS SEJAM EXECUTADAS DENTRO DE PRAZOS ESPECÍFICOS E DE ACORDO COM SUAS PRIORIDADES.



- Prioridade 1 -> TaskRegras -> Lógica da aplicação
- Prioridade 2 -> TaskSensores -> Leitura dos sensores
- Prioridade 3 -> TaskComm -> Comunicação para o monitoramento remoto
- Prioridade 4 -> TaskDisplay -> Exibição das informações no display

CONCEITOS DE ESCALONAMENTO EM SISTEMAS DE TEMPO REAL



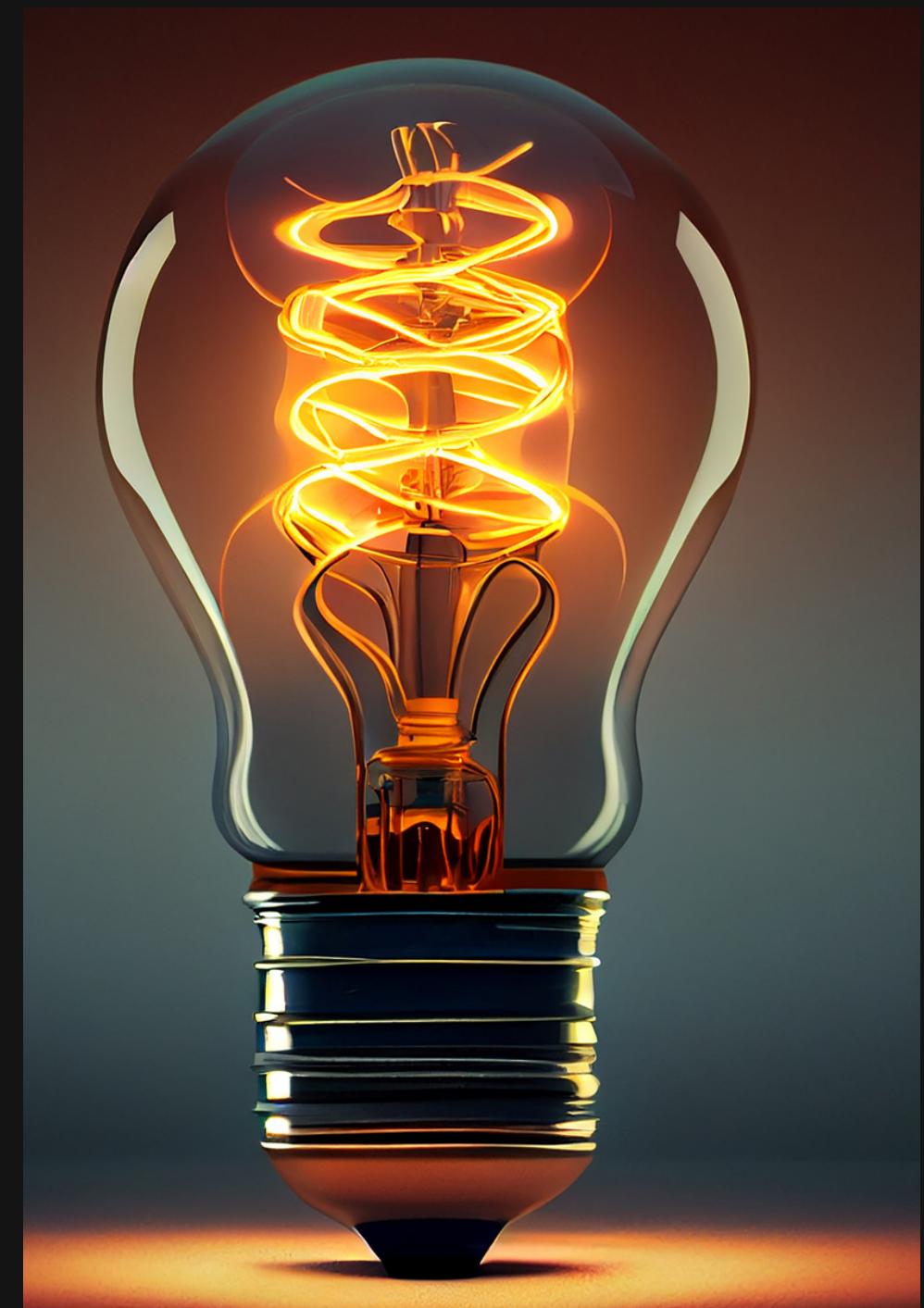
PRIORIDADE: DETERMINA SUA ORDEM DE EXECUÇÃO;

TESTE DE ESCALONABILIDADE: VERIFICA SE OS PROCESSOS SÃO ESCALONÁVEIS, SENDO POSSÍVEL ATENDER A TODOS OS PRAZOS;

TEMPO DE COMPUTAÇÃO (ci): É O TEMPO NECESSÁRIO PARA QUE UMA TAREFA OU PROCESSO SEJA EXECUTADO COMPLETAMENTE;

TEMPO DE INÍCIO (st): REFERE-SE AO MOMENTO EXATO EM QUE UM PROCESSO COMEÇA A SER EXECUTADO APÓS A SUA ATIVAÇÃO;

PERÍODO (pi): É O INTERVALO DE TEMPO ENTRE AS ATIVAÇÕES CONSECUTIVAS DE UM PROCESSO PERIÓDICO.

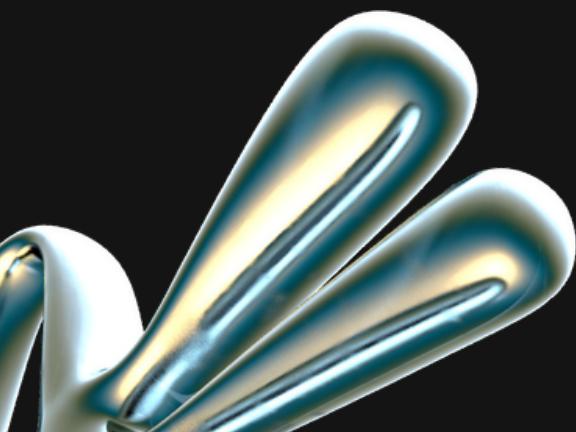
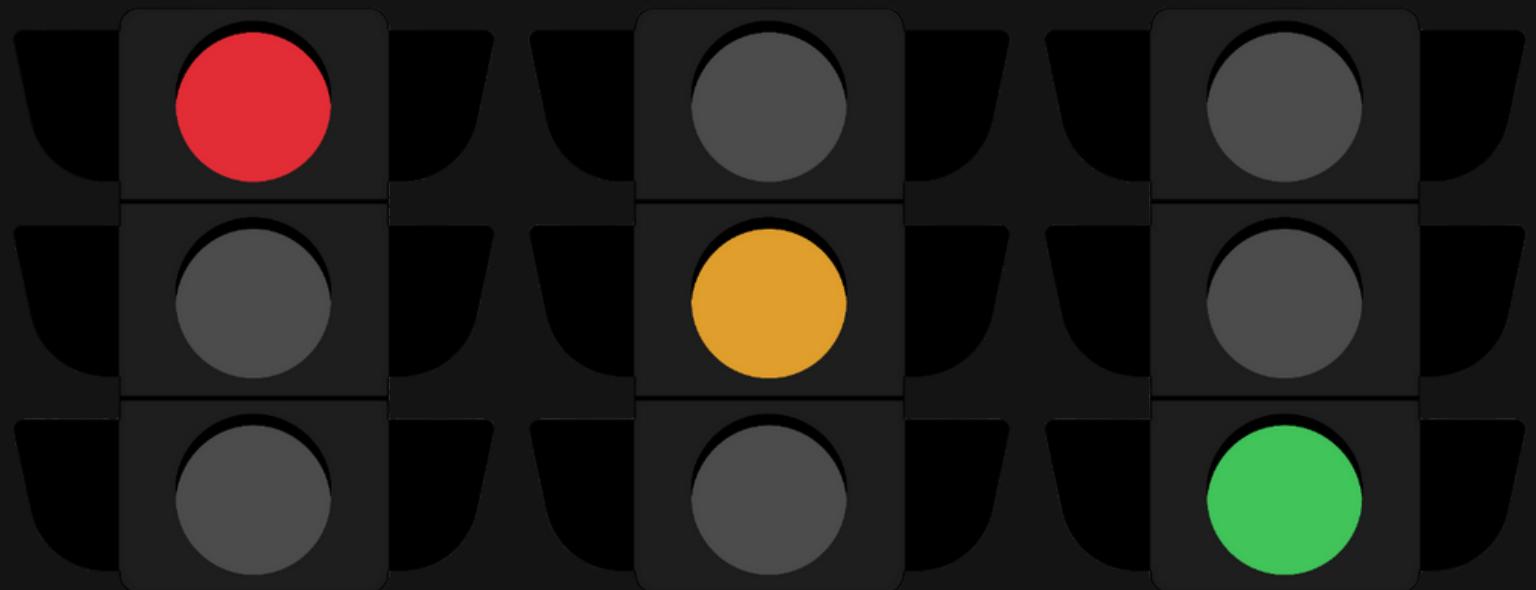


ESCALONAMENTO ONLINE:

- Decisões em Tempo Real;
- Dinâmico;
- Maior Complexidade;
- Adequado para Sistemas Altamente Dinâmicos.

ESCALONAMENTO OFFLINE:

- Decisões Antecipadas;
- Estático;
- Menor Complexidade;
- Adequado para Sistemas Estáveis.



ALGORITMOS DE ESCALONAMENTO EM SISTEMAS DE TEMPO REAL

SÃO MÉTODOS DE RESOLUÇÃO DE ESCALONAMENTO. CLASSIFICADOS COM BASE EM SUA CAPACIDADE DE ATENDER A RESTRIÇÕES DE TEMPO E PRIORIDADES.



ALGORITMOS DE ESCALONAMENTO EM SISTEMAS DE TEMPO REAL

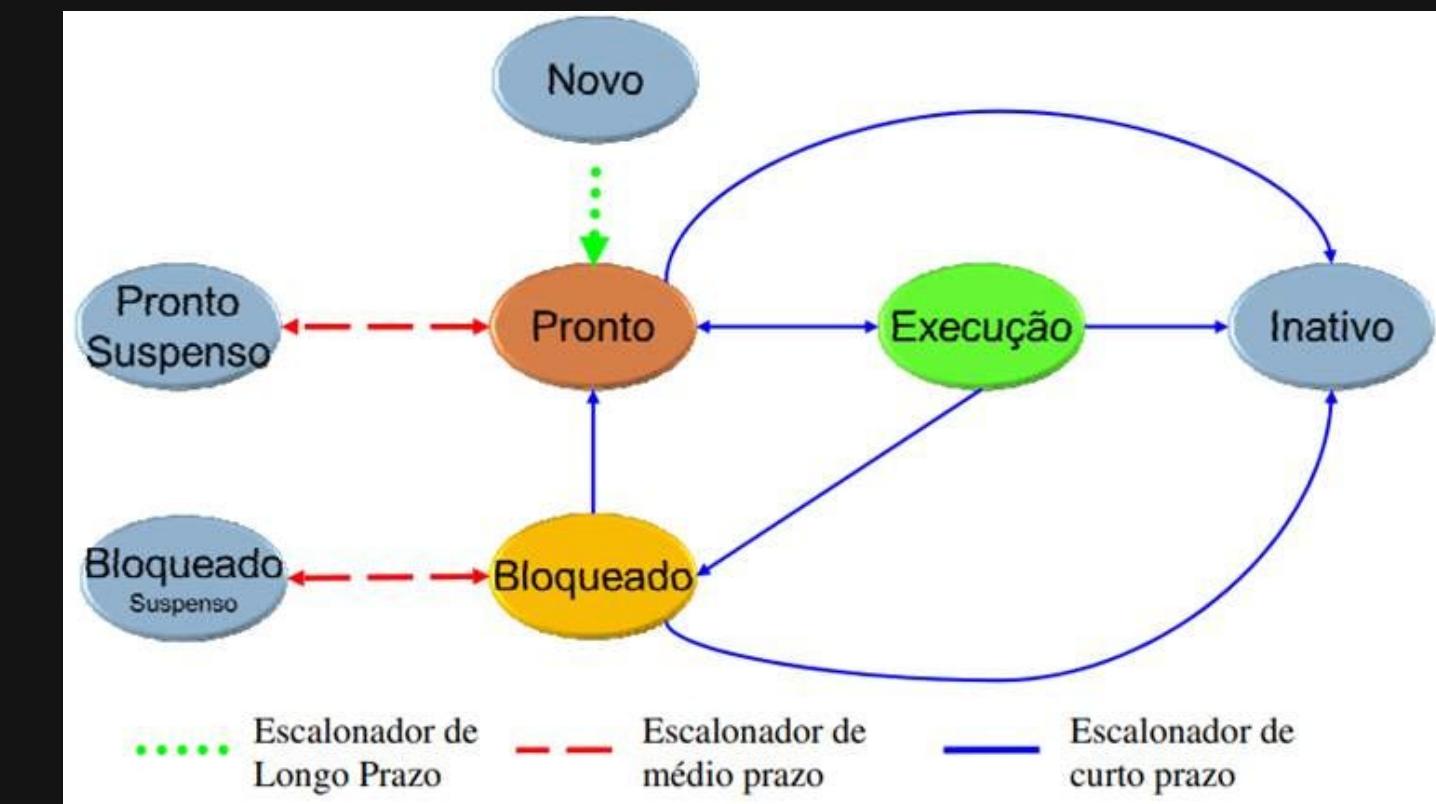


1. PREEMPTIVOS VS. NÃO-PREEMPTIVOS:

- Preemptivos: permitem interromper a execução de um processo em andamento.
- Não-Preemptivos: um processo é executado até a conclusão antes que outro processo possa ser executado.

2. ESTATICAMENTE VS. DINAMICAMENTE:

- Estaticamente: Têm prioridades definidas antecipadamente e não mudam durante a execução.
- Dinamicamente: Os algoritmos de escalonamento dinâmico podem ajustar as prioridades durante a execução com base nas condições em tempo real.



ALGORITMOS DE ESCALONAMENTO EM SISTEMAS DE TEMPO REAL

Rate-Monotonic Scheduling (RMS)

Earliest Deadline First (EDF)

Deadline Monotonic Scheduling (DMS)

Least Laxity First (LLF)

Fixed-Priority Scheduling

Feasibility-Based Scheduling

Round Robin (RR)

Priority Ceiling Protocol (PCP)





NÃO HÁ UM ÚNICO ALGORITMO DE ESCALONAMENTO DE SISTEMAS DE TEMPO REAL QUE SEJA O MAIS IMPORTANTE EM TODOS OS CONTEXTOS. A ESCOLHA DO ALGORITMO DEPENDE DAS NECESSIDADES E RESTRIÇÕES ESPECÍFICAS DE UM SISTEMA EM PARTICULAR.

Rate-Monotonic Scheduling (RMS): amplamente utilizado e eficaz para sistemas com tarefas de período fixo.

Earliest Deadline First (EDF) é conhecido por sua capacidade de otimização em situações mais dinâmicas.

Priority Ceiling Protocol (PCP) é fundamental para sistemas com múltiplos processos concorrentes que compartilham recursos críticos.



EARLIEST DEADLINE FIRST (EDF)



- "PRIMEIRO PRAZO MAIS PRÓXIMO"
- O PRINCIPAL OBJETIVO DO EDF É GARANTIR QUE AS TAREFAS SEJAM EXECUTADAS DE FORMA QUE OS PRAZOS MAIS PRÓXIMOS SEJAM ATENDIDOS PRIMEIRO
- PRIORIDADE DINÂMICA
- ADAPTAÇÃO A PRAZOS VARIÁVEIS
- MENOR PRAZO EM PRIMEIRO LUGAR
- FLEXIBILIDADE
- COMPLEXIDADE
- UTILIZAÇÃO EM APLICAÇÕES CRÍTICAS



EARLIEST DEADLINE FIRST (EDF)



Exemplo:

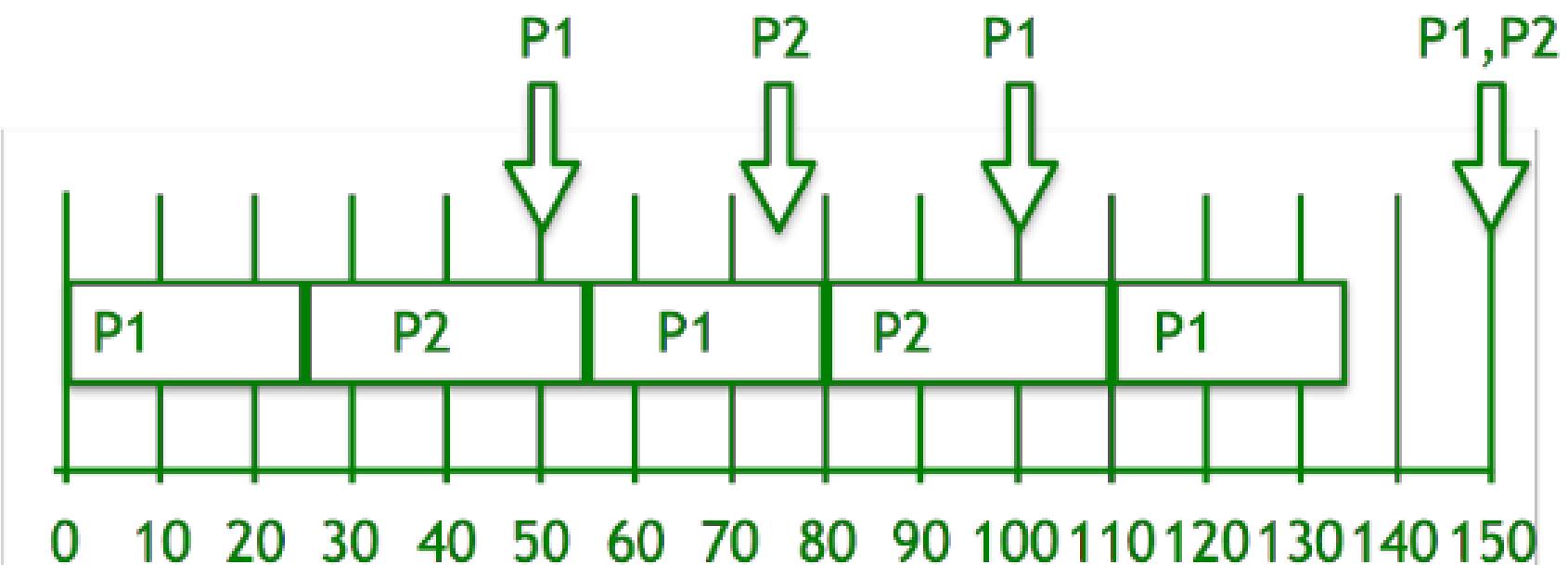
considere dois processos P1 e P2.

Seja o período de P1 $p_1 = 50$

Seja o tempo de processamento de P1 $t_1 = 25$

Seja o período de P2 o período $p_2 = 75$.

Seja o tempo de processamento de P2 $t_2 = 30$



Etapas para solução:

1. O prazo final pf P1 é anterior, portanto, a prioridade de P1 > P2.
2. Inicialmente P1 roda e completa sua execução de 25 vezes.
3. Após 25 vezes, P2 passa a executar até 50 vezes, quando P1 consegue executar.
4. Agora, comparando o deadline de (P1, P2) = (100, 75), P2 continua em execução.
5. P2 conclui seu processamento no tempo 55.
6. P1 passa a executar até o tempo 75, quando P2 pode executar.
7. Agora, novamente comparando o deadline de (P1, P2) = (100, 150), P1 continua a executar.
8. Repita as etapas acima ...
9. Finalmente, no tempo 150, P1 e P2 têm o mesmo deadline, então P2 continuará a executar até seu tempo de processamento após o qual P1 começa a executar.

EARLIEST DEADLINE FIRST (EDF)



Nesse cenário, o escalonamento é:

- Preemptivo: O sistema permite a interrupção de processos em execução;
- Prioridade: A prioridade de P1 é maior do que a de P2 devido ao prazo final (deadline) mais próximo;
- Estático/Dinâmico: Como os prazos finais não mudam durante a execução, é um escalonamento estático;
- Deadlines: O deadline de P1 é 100 e o de P2 é 75;
- Impacto do Deadline na Prioridade: O deadline impacta na prioridade, uma vez que os processos são priorizados com base na proximidade de seus prazos finais. P2 possui um prazo mais curto, o que lhe dá prioridade mais alta em relação a P1.

VANTAGENS

UTILIZAÇÃO EFICIENTE DE RECURSOS

PRAZO DE CONCLUSÃO MAIS CURTO

FLEXIBILIDADE

BOA LATÊNCIA DE RESPOSTA

DESVANTAGENS

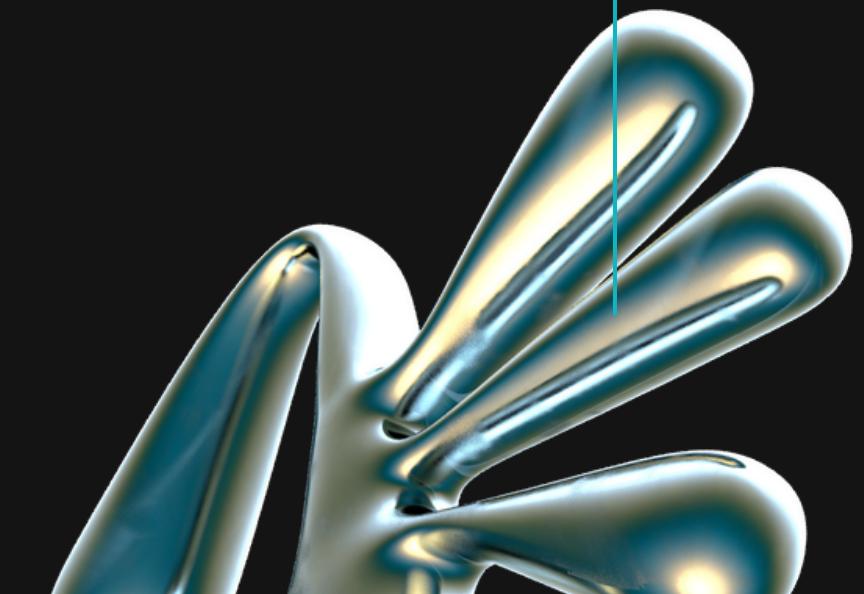
FALTA DE PREVISIBILIDADE

CONDIÇÕES DE SOBRECARGA

COMPLEXIDADE DA IMPLEMENTAÇÃO

IMPACTO DA ALTERAÇÃO DE PRAZOS

NÃO LEVA EM CONSIDERAÇÃO A CARGA DE TRABALHO



Sistemas de tempo real são sistemas computacionais que possuem restrições especiais relacionadas a seu comportamento temporal [Liu 2000].



Liu é autora do livro **Real-Time Systems** (Prentice-Hall, 2000) e co-autora com CL Liu de **Linear Systems Analysis** (McGraw-Hill, 1975).

CONCLUSÃO

Durante o processo de pesquisa bibliográfica, o grupo buscou embasamento teórico em diversas fontes para a elaboração da apresentação. Ao decorrer da apresentação pudemos perceber como funciona o Gerenciamento de Processos.

Através da pesquisa realizada pelos alunos, foi possível perceber como ocorre o funcionamento de um Sistema Operacional e a forma como atua em baixo nível. Com o decorrer do processo de pesquisas, foi verificado os diferentes tipos de processos de escalonamento e suas categorias.

Diante disso, é importante ressaltar, como as pesquisas para a elaboração do seminário foram de extrema importância para o entendimento da forma como os Sistemas Operacionais atuam, e para obter domínio do conteúdo ministrado na apresentação.

REFERÊNCIAS

DEITEL, H. M., DEITEL, P.J., CHOFINES, D.R. Sistemas Operacionais. São Paulo : Pearson Prentice-Hall, 2005.

<https://embarcados.com.br/sistemas-operacionais-de-tempo-real-rtos/>

LEITE, Andreza. Sistemas Operacionais. Disponível em:
<<http://www.univasf.edu.br/~andreza.leite/aulas/SO/ProcessosEscalonamento.pdf>.> Acesso em 6 de novembro de 2023.

Oficina da net. Sistemas Operacionais - O que é Escalonamento de Processos?.2015. Disponível em <http://www.inf.pucrs.br/>. Acesso em 6 de novembro de 2023.

OLIVEIRA, Rômulo Silva de. Fundamentos de Sistemas de Tempo Real. 1. ed. Florianópolis: Kindle Direct Publishing, 2018. ISBN 9781728694047.

SILBERSCHATZ, A., GAGNE, G., GALVIN, P. B. Sistemas Operacionais com Java: conceitos e aplicações. Rio de Janeiro : Campus, 2004.

TANENBAUM, A. S. Sistemas Operacionais Modernos. São Paulo : Prentice-Hall, 2004.

TANENBAUM, A. S., WOODHULL. Sistemas Operacionais: projeto e implementação. 2a. ed.. Porto Alegre : Bookman, 2000.

https://en.wikipedia.org/wiki/Jane_Liu

**FIM DA
APRESENTAÇÃO**

@autor: JOÃO PEDRO AZEVEDO VERAS DOS REIS, LUIS GUILHERME FREITAS DE ALMEIDA SILVA, RENATA COSTA ROCHA CARVALHAL CAVALCANTE

@data última versão: 10/12/2023

@versão: 1.0

@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-lo, 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>.