

Sistemas operacionais: Conceitos e Mecanismos

Capítulo 6: Escalonamento de tarefas

Integrantes do grupo: Jefferson Botitano e Leonardo Ludvig
Professor: Arliones Stevert Hoeller Junior
Disciplina: Sistemas operacionais

Tipos de tarefas

Para definir os algoritmos de escalonamento de tarefas é necessário definir e conhecer bem a natureza das tarefas que um sistema irá executar, podemos classificar o comportamento temporal de uma tarefa em:

- **Tarefas de tempo real:** São tarefas que exigem previsibilidade em seu tempo de resposta a comando externos, geralmente associados ao controle sistemas críticos como automóveis e processos industriais.
- **Tarefas interativas:** São tarefas que necessitam de uma rápida resposta a eventos externos, mas sem a necessidade da previsibilidade de tarefas de tempo real, envolvendo mais trocas de contexto nas suas aplicações.
- **Tarefas em lote (batch):** Exigem maior tempo de processamento e pouca necessita de resposta a eventos externos, sendo geralmente grandes massas de dados, backups, renderizações, cálculos numéricos longos que executam sem a interação com o usuário.

Também podemos classificar as tarefas de acordo com o seu comportamento ao longo do seu uso no processador, como:

- **Tarefas orientadas a processamento:** Passam a maior parte de sua existência no estado de pronta ou executando necessitando intensivamente de processamento.
- **Tarefas orientadas a entrada/saída:** Utilizam mais os dispositivos de entrada e saída do que o processador, passando maior parte do tempo suspensas aguardando as solicitações de leitura ou escrita,

Objetivos e métricas

Para o desenvolvimento de escalonadores eficientes é necessário ter em mente para qual objetivo ele será desenvolvido, levando em consideração quais tipos de tarefa (interativas, de lote) ele irá gerenciar e quais critérios serão levados em consideração para obter bons resultados como menor tempo de processamento e de espera das tarefas a serem executadas, alguns critérios que podem ser levados em consideração são:

- **Tempo de execução:** Tempo total de existência da tarefa, desde a sua criação, processamento e finalização, contando também seu tempo em filas de espera.
- **Tempo de espera:** É o tempo contabilizado quando a tarefa está na fila de pronta aguardando sua vez de ser processada, não contando seu tempo de espera para em operações de entrada/saída.
- **Tempo de resposta:** Utilizado em sistemas interativos, é o tempo de resposta da chegada de um evento e o seu processamento, como o tempo de clicar no mouse e o click ser efetuado no monitor.
- **Justiça:** Tarefas de prioridade e comportamento similares devem ter tempo de execução similares.
- **Eficiência:** Indica a eficiência das trocas de contexto, ou seja, o grau de utilização do processador na execução de tarefas de usuário.

Escalonamento preemptivo e cooperativo

Podemos dividir os escalonadores em dois grupos em relação ao seu gerenciamento de execução de tarefas, sendo chamados de escalonadores preemptivos ou cooperativos. Nos algoritmos cooperativos uma tarefa que está em processamento só sairá desse estado quando terminar de ser processada, libere explicitamente o processador ou solicite uma operação de entrada/saída, não havendo interrupções, o que leva a necessidade das tarefas cooperarem entre si para um bom desempenho. Nos sistemas atuais é muito mais utilizado o escalonador preemptivo, onde há o uso de troca de contexto por meio de esgotamento de quantum, chamadas de sistemas ou até mesmo interrupções que chamem tarefas de maior prioridade. A cada troca de contexto o escalonador reavalia a prioridade de todas as tarefas para decidir qual a próxima a ser executada.

Algoritmos de escalonamento de tarefas

- **First-Come, First Served(FCFS):** escalonamento referente a tarefas em sequência, a tarefa é executada à medida que elas estão prontas para serem executadas e conforme a sua ordem de chegada na fila de tarefas prontas.
- **Round-Robin(RR):**escalona as tarefas dentro de um laço,realizando uma execução circular das tarefas e distribuindo melhor uso do processador entre as tarefas ao longo do tempo.
- **Shortest Job First(SJF):**escalonamento de tarefas de acordo com a tarefa que irá usar por menos tempo o processador dentre aquelas que estão na fila de tarefas prontas.
- **Shortest Remaining Time First(SRTF):**Sendo um algoritmo cooperativo que ao receber uma tarefa ao processador ela recebe e executa até encerrar,compara a duração prevista de cada nova tarefa e ao receber uma nova tarefa e ela possuir um tempo restante menor, ela irá ser recebida ao processador.
- **Escalonamento por prioridades Fixas(PRIOc):**O escalonamento por prioridade é baseado em que cada tarefa é associada uma prioridade representados por um valor inteiro e que são usados para definir uma ordem de execução das tarefas e que no caso de ser cooperativo os valores de prioridade são feitos em escala de prioridade positiva sendo aqueles de maior valor numérico representando maior prioridade de execução.
- **Escalonamento por prioridades Fixas(PRIOp):**escalonamento identifica uma tarefa de maior prioridade estar disponível para execução o processo é entregá-lo ao processador e quanto a tarefa que estava sendo executada volta para fila de tarefas prontas repetindo o processo e sempre buscando uma tarefa para executar de maior prioridade.
- **Escalonamento por prioridades dinâmicas (PRIOd):** algoritmo que consertar a não execução de tarefas que possuem menor prioridade, que quando uma tarefa atingir um tempo de não ter sido executada ela recebe um nível de prioridade para poder ser executada.

Explique o que é, escalonamento round-robin, dando um exemplo.

Popularmente conhecido como escalonamento por revezamento, é um algoritmo preemptivo que pode levar em consideração alguns fatores de troca de contexto como o esgotamento do quantum da tarefa ou seu tempo de envelhecimento, realizando assim uma troca da tarefa a ser executada pela próxima na fila de tarefas prontas, porém se o número de trocas de contexto for muito alto pode comprometer o desempenho do escalonador.

Um exemplo temos o gráfico onde a tarefa 1 é executada e em seguida interrompida pois a tarefa 2 será ser executada e quando finalizada a seguinte tarefa a chegar ao processador será a de número 3 que sofre interrupção por ter quantum de valor 2 segundos. A tarefa 1 que havia sido interrompida retorna a execução e executada por 2s sendo o quantum da tarefa 3 sendo interrompida novamente para dar espaço a tarefa 4 que é executada e finalizada cedendo vaga para tarefa 5 ser executada e finalizada, como a tarefa 3 havia sido interrompida ela retorna sua execução e termina e por último a tarefa 5 também volta para o processador para poder ser executada e finalizada.

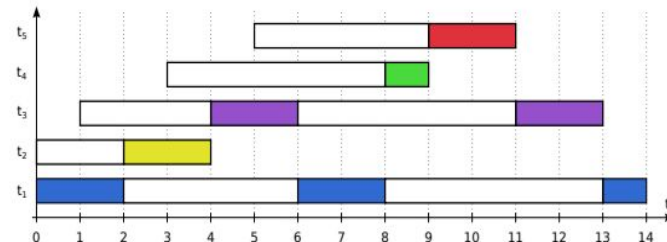


Figura 6.2: Escalonamento Round-Robin.

Explique o que é, para que serve e como funciona a técnica de aging.

A técnica de aging surgiu como uma solução para algoritmos escalonadores onde a prioridade ou tempo de execução de uma tarefa é levada como principal critério para escolha de processamento, tornando tarefas de baixa prioridade ou tempo de execução muito longo incapazes de serem processadas em sistemas com alta demanda de tarefas de maior prioridade e menor tempo de processamento. Essa solução conta com mais um critério para definir a próxima tarefa a ser executada, o qual seria o seu tempo de espera na fila de pronta, ou seja a cada ciclo essa prioridade é incrementada, aumentando a sua prioridade e eliminando o problema de inanição das tarefas.

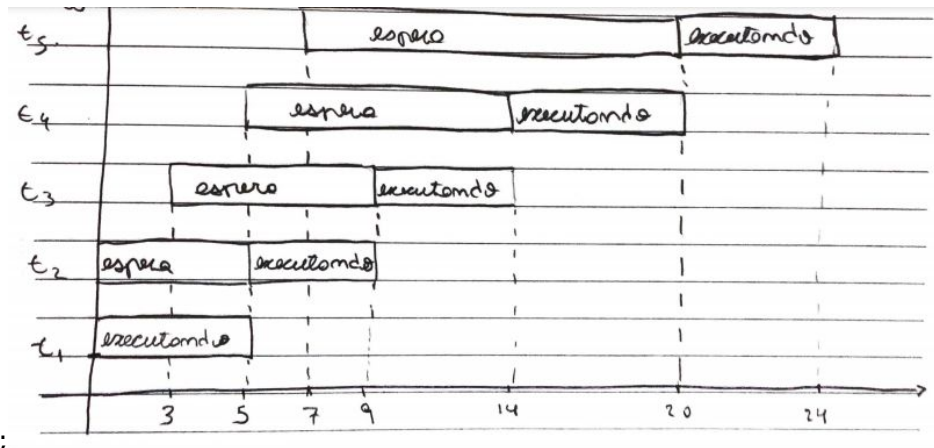
A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(a) FCFS cooperativa

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.



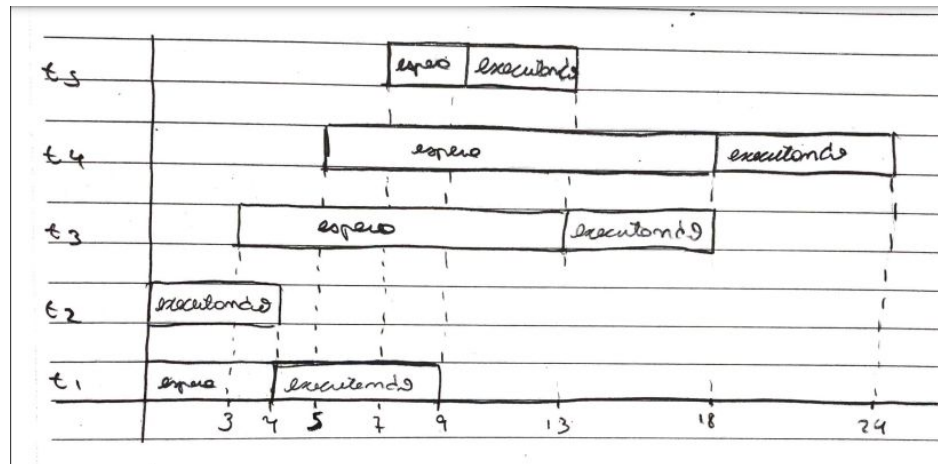
A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(b) SJF cooperativa

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.



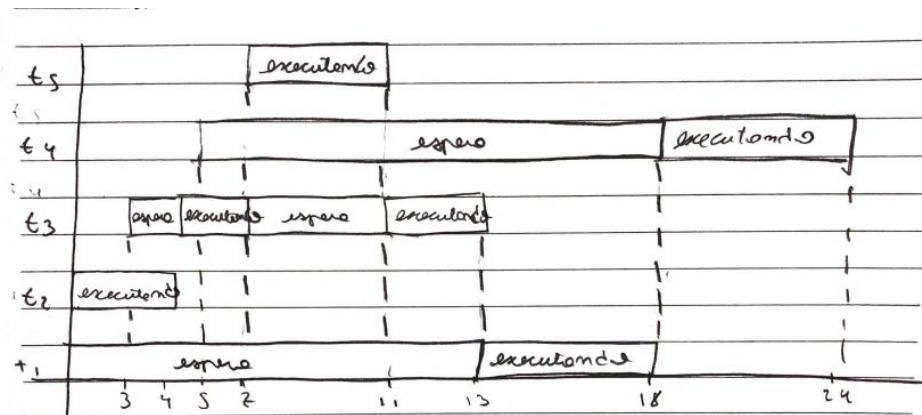
A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(c) SJF preemptiva (SRTF)

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.



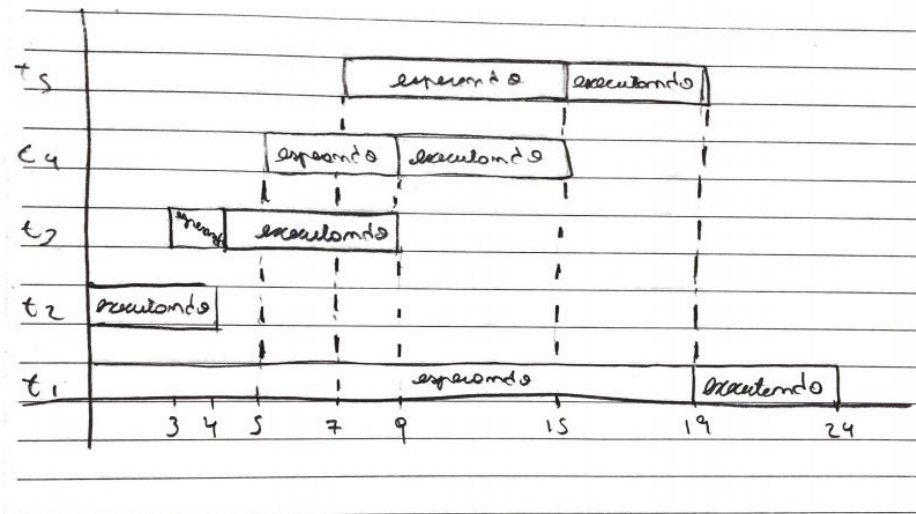
A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(d) PRIO cooperativa

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.



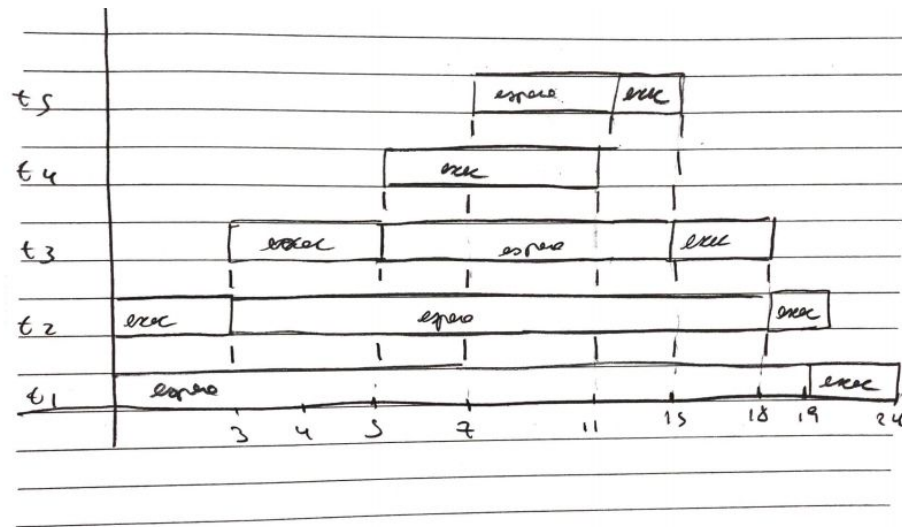
A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(e) PRIO preemptiva

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.



A tabela a seguir representa um conjunto de tarefas prontas para utilizar um processador:

Tarefa	t_1	t_2	t_3	t_4	t_5
ingresso	0	0	3	5	7
duração	5	4	5	6	4
prioridade	2	3	5	9	6

Represente graficamente a sequência de execução das tarefas e calcule os tempos médios de vida (turnaround time) e de espera (waiting time), para as políticas de escalonamento a seguir:

(f) RR com quantum = 2, sem envelhecimento

Considerações: todas as tarefas são orientadas a processamento; as trocas de contexto têm duração nula; em eventuais empates (idade, prioridade, duração, etc), a tarefa com menor índice prevalece (i.e., t_1 prevalece sobre t_2 , que prevalece sobre t_3 , etc); valores maiores de prioridade indicam maior prioridade.

