

ESCALONAMENTO DE PROCESSOS

APS de sistemas operacionais

Carolina Ramos de Souza Pinto, Daniel Moraes, Nicoly Alves Saguini, Rhaissa Vilela, Gabriela de Souza Pereira

Análise e desenvolvimento de sistemas – 2º semestre, matutino
Centro Universitário Das Faculdades Metropolitanas Unidas (FMU)
São Paulo – SP - Brasil

Resumo. *O escalonamento de processos é o gerenciamento da ordem em que processos são executados em um sistema operacional. Essa tarefa é feita para maximizar a eficiência do sistema e garantir que os tais recursos sejam usados de maneira justa pelos diferentes processos. Existem vários algoritmos de escalonamento disponíveis, cada um com suas próprias vantagens e desvantagens. A escolha do algoritmo depende das necessidades do usuário e das características do sistema. O escalonamento é fundamental para garantir que o sistema operacional funcione de forma ágil, eficiente, prática e estável, permitindo que os usuários realizem suas tarefas com rapidez e segurança.*

1. Introdução

O escalonamento de processos é uma função crítica dos sistemas operacionais que gerenciam a alocação de recursos do sistema para múltiplos processos em execução. Em um sistema multitarefa, vários processos podem estar em execução simultaneamente e competindo pelos recursos do sistema, como CPU, memória, E/S de disco, entre outros. O escalonador de processos é responsável por decidir qual processo será executado em seguida e por quanto tempo, garantindo um equilíbrio justo e eficiente na utilização desses recursos.

O objetivo do escalonamento de processos é maximizar a utilização da CPU, minimizar o tempo de resposta do usuário e garantir a equidade no acesso aos recursos do sistema para todos os processos em execução. Existem diferentes algoritmos de escalonamento que podem ser usados para alcançar esses objetivos, e a escolha do algoritmo depende das características do sistema, como o número de processadores, o tipo de aplicação em execução e as prioridades dos processos.

Em resumo, o escalonamento de processos é uma parte essencial dos sistemas operacionais modernos e é responsável por manter o sistema operando de maneira eficiente e equilibrada, garantindo que os recursos do sistema sejam utilizados de forma justa e adequada.

2. Conceito de escalonamento de sistemas

É uma técnica utilizada pelos sistemas operacionais para gerenciar a execução de múltiplos processos concorrentes em um ou mais ocorridos. O objetivo principal do escalonamento é garantir a utilização eficiente dos recursos dos sistemas, como por exemplo o processador, memória e dispositivos de entrada/saída.

Existe vários algoritmos de escalonamento de processos, cada um com as suas próprias características e vantagens. Alguns dos conceitos fundamentais do escalonamento de processos incluem:

- (1) Prioridade: cada qual processo é atribuído uma prioridade com base em seu nível de importância ou necessidade de recursos do sistema. Processos com maior prioridade são escalonados antes dos processos com menor prioridade;
- (2) First Come: Escalona os processos na ordem em que foram submetidos ao sistema operacional;
- (3) Shortest Job First: Prioriza a execução dos processos com menor tempo de execução, podendo resultar em menor tempo de espera e maior vazão do sistema.
- (4) Round Robin: Permite a execução concorrente de processos, onde cada processo recebe um pequeno intervalo de tempo para execução (chamado de quantum) e, após esse tempo, é interrompido para que outro processo possa executar, e assim por diante.

Esses algoritmos de escalonamento também podem apresentar diferentes características como a perceptividade, que permite a interrupção de um processo em execução para a execução de outro processo com maior prioridade, ou a não perceptividade, onde um processo em execução só é interrompido quando completa sua execução ou libera voluntariamente o processador.

Outros critérios de seleção de processos para escalonamento incluem a implementação de filas de espera para processos prontos para execução, o cálculo do tempo de espera dos processos e a implementação de políticas de envelhecimento para evitar a inanição de processos com menor prioridade.

3. Algoritmos de escalonamento

Para dar a ilusão em que cada tarefa/programa está sendo executada de uma só vez, o processador/CPU executa somente um programa e trabalha de forma tão rápida, trocando a ocupação do processador, que dá a ilusão que está executando vários processos de uma só vez. O mecanismo responsável por fazer essas trocas de processos é chamado de multiprogramação; em que, paralelamente, existe diversos programas aguardado para utilizar a CPU.

Responsável por decidir qual processo irá ocupar o CPU, está o escalonador de processo. Para isso, é utilizado algoritmos que fica responsável para a tomada de decisões com vários critérios de decisão. Para decidir qual escalonador utilizar em um Sistema Operacional, é necessário avaliar em qual situação em que o SO será utilizado para o processamento de dados.

Existe, também, os tipos de comportamento dos processos em que o CPU precisa lidar; os orientados a Entrada e Saída (IN/OUT) e os orientados a CPU. Isso significa que comportamento IN/OUT não exige maior processamento do processador ou ocupação em que esse tipo de programa, recebe um parâmetro de entrada e logo retorna o resultado. Os processos orientados a CPU, os programas está constantemente utilizando o espaço que a CPU disponibiliza. Exemplo: programas de texto em que existe a entrada de dados via teclado e retorno dos dados “instantâneo” dos dados digitados e exige que este processo entre e saia da CPU.

3.1 Categoria de Algoritmos de Escalonamento de Processos

Existem cenários que possibilita a troca de ocupação dos processos na CPU. A classificação de diferentes lógicas para qual tipo de algoritmo irá atuar para gerenciar os processos. São eles;

- (1) Sistemas em Lote; maximiza o número de “Jobs” por hora, ou seja, processa várias tarefas reunidas, minimiza o tempo entre submissão e término e mantém a CPU sempre ocupada.
- (2) Sistemas interativos; tempo de resposta, responde rapidamente todas as requisições, proporcionalidade.
- (3) Sistema de tempo real; possui o objetivo de cumprir prazos, evitar perda de dados, evita degradação dos sistemas multimídia.

3.2 Algoritmos de base conceitual.

Algoritmos que servem para a construção de algoritmos mais complexos são;

- (1) FCFS – First Come, First Served.

Consiste em processar as tarefas de forma sequencial quando elas tornam-se prontas. Sua vantagem está sobre sua simplicidade de executar as tarefas.

- (2) RR – Round Robin

Como dito anteriormente, os algoritmos de escalonamento sofrem variações a partir de 1 algoritmo. Nesse caso, o FCFS da origem ao Round Robin que funciona por revezamento. Ou seja, reparte o CPU entre todos os processos prontos para execução. Os processos são organizados em uma fila circular, guardando para cada execução uma fatia de tempo da CPU, em que, se o processo não terminar dentro de seu tempo que lhe foi reservado, ele é colocado no fim da fila e uma nova fatia de tempo é reservada para ele.

O escalonamento circular pode ser simples, mas traz problemas quando os tempos que foram reservados para alguns processos é muito alto comparado a outros. Se existir tarefas de longa duração no sistema, as tarefas curtas terão seu tempo de resposta lento, pois as tarefas longas ficarão rodando por muito mais tempo na CPU.

- (3) Shortest Job First

Este algoritmo executa primeiro as tarefas menores, ou seja, o espaço do processador é reservado para executar as tarefas de menor tempo de execução. Proporciona mais execução de tarefas, no entanto, o problema deste algoritmo é que houver o fornecimento constante de tarefas curtas, o algoritmo não executará as tarefas de tempo médio longo. Ou seja, este algoritmo falha em dar prioridades para as tarefas.

(4) Shortest Remaining Time First

Algoritmo cooperativo, pois uma vez que a tarefa recebe o espaço do processador, ela executa esta tarefa até encerrar. A vantagem deste escalonador é que o processador só irá se preocupar em executar uma tarefa quando a que já está em execução estiver pronta.

(5) Escalonamento por prioridades fixas - PRIOc, PRIOp

Existem múltiplos critérios para ordenar a fila de tarefas prontas e escolher a próxima tarefa, pois ela depende do valor associado ao processo. Exemplos de mudanças que pode mudar a prioridade da tarefa: Comportamento da tarefa, prioridade em que varia de acordo com o valor associado ao processo, grau de interação etc. Quando a tarefa de maior valor associado se torna disponível para execução, este algoritmo disponibiliza o processador a esta tarefa e traz a tarefa que está em execução de volta para a fila de tarefas prontas.

(6) Escalonamento de Prioridades Fixas – PRIOd

No Shortest Job First, o processador que tiver maior tempo de processamento pode a nunca executar, para evitar este problema denominado de “Starvation” só foi resolvido quando um outro algoritmo foi desenvolvido para acabar com essa necessidade. Portanto, o Escalonamento de Prioridades Fixas surgiu com essa premissa, para evitar a “inanição” do algoritmo anterior. Ele funciona com o aumento “envelhecimento” de uma tarefa de baixo tempo de execução, assim, este processo aumenta o envelhecimento proporcionalmente ao seu tempo que está em execução. Definindo assim, prioridade dinâmica que permite executar aquela tarefa de menor tempo de execução periodicamente. Em cada turno de execução, o processador escolhe a próxima tarefa aquela que tiver a maior prioridade dinâmica. Aquela que tiver o maior fator de envelhecimento, terá mais chance de execução pelo processador.

3.3 Comparação de tempo de execução dos algoritmos de escalonamento;

Algoritmo de escalonamento	FCFS	RR	SJF	SRTF	PRIOc	PRIOp	PRIOd
Tempo médio de execução T_t	8,0	8,4	5,8	5,4	6,6	5,6	5,8
Tempo médio de espera T_w	5,2	5,6	3,0	2,6	3,8	2,8	3,0
Número de trocas de contexto	4	7	4	5	4	6	6
Tempo total de processamento	14	14	14	14	14	14	14

Figura 1 (Maziero, 2019, p.82)

4. Tipos de escalonamento

Escalonador de processos é um subsistema utilizado pelo sistema operacional, para melhorar a performance do processador. Cada processo passa por um tipo de prioridade. Sendo assim, temos o uso do tempo do processador de forma justa, os processos passam por um escalonamento tornando a utilização dos componentes mais eficiente. Com o método de escalonamento de processos, a CPU de forma geral não fica com o tempo nulo sem produzir. Podemos entender processos como um programa, a todo momento a CPU realiza a troca de processos e o escalonamento é responsável por decidir qual processo tem maior prioridade de execução, com base em algoritmo de escalonamento.

Para entender de uma forma mais prática o funcionamento do escalonador, podemos iniciar com a execução de um processo, esse processo tem um tempo x (chamado de quantum) para ser executado, após atingir esse limite de tempo mesmo que não tenha terminado, ele entra na lista do escalonador, porém vai para o fim da lista, ou seja, sua prioridade passou a ser a última. A partir do momento em que um processo é executado o escalonador define sua prioridade, cada processo criado atinge um nível de prioridade definido pelo escalonador. Essa prioridade é definida por dois tipos:

- (1) Hierarquia – quando a prioridade se dá pela ordem da criação do processo.
- (2) Dinâmica – O escalonador verifica a quantidade de vezes que o processo está sendo executado e a quantidade de E/S está ocorrendo. Assim sendo a prioridade passa a ser desse processo que tem uma maior execução.

Vale ressaltar que todo esse processo ocorre em tempo real para que o desempenho final seja mais produtivo. O escalonamento de processos quando cria suas prioridades entrega melhorias em diversos campos como maximizar o tempo do processador, minimizar o tempo de espera e resposta.

5. Critérios de rendimento

Os critérios de desempenho em um escalonamento de processos são utilizados para avaliar o funcionamento do processo e verificar se ele está operando conforme o esperado. Esses critérios incluem o tempo de ciclo, o tempo de espera, o tempo de processamento e o tempo de resposta. O tempo de ciclo engloba todo o período que um processo leva para ser concluído, incluindo o tempo de espera e o tempo de processamento. O tempo de espera é o intervalo que um processo permanece na fila antes de ser executado, enquanto o tempo de processamento é o período em que o processo é executado. O tempo de resposta representa o tempo necessário para o sistema responder a uma solicitação de um processo, compreendendo o tempo de espera e o tempo de processamento. Além desses critérios, outros fatores, como a utilização da CPU e a taxa de transferência de entrada/saída, podem ser considerados para avaliar o desempenho de um processo no escalonamento.

Os critérios de escalonamento são as métricas utilizadas para determinar qual processo será o próximo a receber a CPU. Existem diversos critérios que podem ser

adotados, e cada um apresenta vantagens e desvantagens próprias. Entre os critérios mais comuns, estão:

- (1) Tempo de chegada: o processo que chegou primeiro é executado primeiro.
- (2) Tempo de execução: cada processo é executado por um tempo determinado, após o qual é interrompido e outro processo é executado. Isso pode ser útil para evitar que um processo monopolize a CPU por muito tempo.
- (3) Prazo de entrega: o processo que tem um prazo de entrega mais próximo recebe prioridade.
- (4) Utilização de CPU: o processo que usa menos CPU é executado primeiro.
- (5) Tempo de espera: É o tempo total que o processo permanece na fila de pronto durante seu processamento aguardando para ser executado. A redução do tempo de espera dos processos é desejada pela maioria das políticas de escalonamento.
- (6) Tempo de resposta: o processo que tem um tempo de resposta mais curto é executado primeiro.
- (7) Prioridade: cada processo é atribuído uma prioridade, e o processo com a maior prioridade é executado primeiro. Isso pode ser útil quando certos processos precisam ser executados com urgência.
- (8) Tempo de processador: É o tempo que um processo leva no estado de execução durante seu processamento. As políticas de escalonamento não influenciam o tempo de processador de um processo, sendo este tempo função apenas do código da aplicação, e da entrada de dados
- (9) Fairness: cada processo recebe uma quantidade igual de tempo da CPU, independentemente de sua prioridade ou tempo de espera. Isso pode ser útil para garantir que todos os processos recebam uma quantia justa de tempo de CPU.
- (10) Throughput: representa o número de processos executados e um determinado intervalo de tempo. A maximização do throughput é desejada na maioria dos sistemas.
- (11) Utilização do processador: Na maioria dos sistemas é desejável que o processador permaneça a maior parte do tempo ocupado uma utilização na faixa de 30% indica um sistema com uma carga de processamento baixo enquanto na faixa de 90% indica o sistema bastante carregado próximo de sua capacidade máxima.

6. Bibliografia

- (1) COLETTA, Alex De Francischi. Escalonamento de Processos. São Paulo. Disponível em < <https://alexcoletta.eng.br/artigos/escalonamento-de-processos/> Acesso em: 11/05/2023.
- (2) ALVES, Danilo da Silva. ESTUDO COMPUTACIONAL DE MÉTODOS PARA ESCALONAMENTO DE PROCESSOS COM ABORDAGEM CLÁSSICA E FUZZY VIA SIMULAÇÃO. **Escalonamento**, [s. l.], 3 fev. 2018. Disponível em: https://www.formiga.ifmg.edu.br/documents/2018/Biblioteca/TCCs_e_Artigos/Danilo_da_Silva_Alves.pdf. Acesso em: 11 maio 2023.

- (3) NOVATO, Douglas. Sistemas Operacionais - O que é Escalonamento de Processos?. **Escalonamento**, [s. l.], 22 maio 2014. Disponível em: <https://www.oficinadanet.com.br/post/12781-sistemas-operacionais-o-que-e-escalonamento-de-processos>. Acesso em: 13 maio 2023.
- (4) ESCALONAMENTO de processos. [S. l.], 9 jun. 2016. Disponível em: https://pt.wikipedia.org/wiki/Escalonamento_de_processos. Acesso em: 13 maio 2023.
- (5) LEITE, Andreza. SISTEMAS OPERACIONAIS. **Sistemas operacionais**, [s. l.], 2019. Disponível em: <http://www.univasf.edu.br/~andrea.leite/aulas/SO/ProcessosEscalonamento.pdf>. Acesso em: 13 maio 2023.
- (6) LEITE, Samantha. Algoritmos de Escalonamento de Processos. Nasemanadaprova, 2017. Disponível em: nasemanadaprova.blogspot.com/2015/12/algoritmos-de-escalonamento-de-processos.html. Acesso em: 13 maio, 2023.

7. Referência científica

- (1) SOUZA, Márcio Augusto de. Uma abordagem para a avaliação do escalonamento de processos em sistemas distribuídos baseada em monitoração. **Digital Library USP**, [S. l.], p. 168, 5 ago. 2018. Disponível em: <https://www.teses.usp.br/teses/disponiveis/55/55134/tde-05012018-103739/en.php>. Acesso em: 11 maio 2023.
- (2) Maziero, Carlos. **Sistemas Operacionais: Conceitos e Mecanismos**. UFPR, Curitiba: DINF, 2019.
- (3) MEIRA, Marcos Vinícius de. POLÍTICA DE ESCALONAMENTO DE PROCESSOS EM LINUX. **Linux**, [s. l.], 2008. Disponível em: <https://revista2.grupointegrado.br/revista/index.php/campodigital/article/view/344/159>. Acesso em: 11 maio 2023.
- (4) EUSTÁQUIO, Paulo Sérgio Franco. Estudo das Principais Políticas de Escalonamento de Processos em Sistemas Operacionais Multitarefa Utilizando Técnicas de Modelagem e Simulação Orientada a Eventos. **Escalonamento de Processos**, [s. l.], 2008. Disponível em: https://www.researchgate.net/profile/Paulo-Eustaquio/publication/245025153_Estudo_das_Principais_Politicass_de_Escalonamento_de_Processos_em_Sistemas_Operacionais_Multitarefa_Utilizando_Tecnicas_de_Modelagem_e_Simulacao_Orientada_a_Eventos/links/00b4951d5f7f4c9b59000000/Estudo-das-Principais-Politicass-de-Escalonamento-de-Processos-em-Sistemas-Operacionais-Multitarefa-Utilizando-Tecnicas-de-Modelagem-e-Simulacao-Orientada-a-Eventos.pdf. Acesso em: 11 maio 2023.