

# Sistema Simulador Escalonamento de Processos em Sistemas Operacionais

**Maxwell Anderson Ielpo do Amaral**<sup>1</sup>

Associação Paraibana de Ensino Renovado – ASPER  
Av. Prof. Joaquim Francisco Veloso Galvão,  
nº 1860 - Bairro dos Estados - João Pessoa, PB, Brasil  
Fone: (83) 2106-9500  
E-mail: maxwellcc@hotmail.com

**Estevam Pessoa do Nascimento Santiago**

Associação Paraibana de Ensino Renovado – ASPER  
E-mail: estevampessoa@gmail.com

**Josemary Marcionila Freire dos Santos**

Departamento de Estatística - UFPB  
Universidade Federal da Paraíba - UFPB  
João Pessoa, Brasil, Fone (83) 3216-7075  
E-mail: josemaryfreire@terra.com.br

**Resumo:** *O presente trabalho descreve um simulador de Escalonamento de Processos de um sistema operacional multitarefa moderno que foi projetado inicialmente para avaliação acadêmica semestral. Posteriormente, com a evolução do projeto, acabou se tornando uma ferramenta para a ministração de aulas de Gerenciamento de Processadores, contribuindo para formação profissional do corpo discente de forma visível, prática e rápida. Fundamenta-se em estudos teóricos e pesquisas realizadas em sala de aula em um curso de Ciência da Computação, discorrendo inicialmente a uma introdução da definição dos escalonadores e seus critérios de escalonamento. Também descreve os objetivos do projeto e o funcionamento do simulador. Ao fim apresentamos de maneira concisa os resultados obtidos em uma simulação de execução de processos em um escalonamento tipo FIFO (First In, First Out).*

**Palavras Chave:** *simuladores, escalonadores, processos, sistemas operacionais, gerenciamento.*

## 1. Introdução

Os computadores modernos possuem uma característica interessante ao aspecto dos usuários comuns que é aquela de onde se percebe que vários aplicativos estão sendo executados simultaneamente. Na verdade, os aplicativos, ou processos do sistema operacional, somente podem ser exclusivamente executados por vez em sistemas computacionais que possuem um único processador ou UCP. Os escalonadores permitem que vários processos sejam executados concorrentemente pelo uso de um único processador, trazendo vários benefícios aos usuários e melhorando a interoperabilidade em sistemas.

O escalonamento de processos em um sistema operacional possui diversas funções, como: “manter o processador ocupado a maior parte do tempo, balancear o uso da UCP entre os processos, privilegiar a

execução de aplicações críticas, maximizar o *throughput* do sistema e oferecer tempos de respostas razoáveis para usuários interativos” (TANENBAUM, 2003)

O escalonador é uma rotina do sistema operacional que tem por função implementar os critérios da política de escalonamento. Os principais critérios que devem ser considerados em uma política de escalonamento são aqueles que visam a uma maior utilização da UCP, maior número de processos executados em um determinado intervalo de tempo, uma melhor redução do tempo de espera dos processos na fila de pronto, menor tempo de *turnaround*, que é o tempo total que um processo permanece em um sistema desde a sua criação até que entre em estado de término, e um menor tempo dentre uma requisição ao sistema ou à aplicação e o instante em que a resposta é exibida na tela, etc.

Este simulador foi criado com o objetivo de demonstrar interativamente como os processos são

<sup>1</sup> Autor a quem toda correspondência deverá ser endereçada.

escalonados em um sistema, donde poderá ser escolhido o tipo de escalonamento a ser simulado. Os dados de entrada para os processos, como o tempo de UCP, a prioridade e o *timeslice* ou fatia de tempo, se forem os casos, podem ser obtidos com entrada de informações feita pelo usuário, ou o simulador disponibilizará estes dados automaticamente.

Durante a simulação, poderemos observar os processos percorrendo as filas de pronto e de espera, sendo processados no estado de execução, até chegarem ao seu término. Em tempo real, também poderemos observar o preenchimento do gráfico que demonstra o uso dos processos pela UCP de acordo com um instante de tempo. Este simulador é ideal para submeter a teste os conhecimentos adquiridos no estudo do Gerenciamento dos Processadores e facilitar a ministração das aulas feitas pelos professores de Ciência da Computação e outros cursos.

## 2. Objetivos

- **Geral**

Desenvolver um sistema informatizado simulador de escalonamento de processos em sistemas operacionais multitarefa.

- **Específicos**

1. Identificar as principais características dos processos e dos processadores;
2. Conhecer os principais escalonadores de processos implementados pelos projetistas de sistemas operacionais modernos;
3. Criar algoritmos que permitam simular os escalonadores mais utilizados pelos projetistas e avaliados pelos principais pesquisadores da área;
4. Projetar e desenvolver o simulador com o intuito de verificar resultados de escalonamentos aplicados a uma determinada quantidade e tipos de processos diferentes sobre UCP's com frequências variadas de operação;
5. Verificar a performance e o melhor tempo de resposta dos processamentos escalonados em uma UCP;
6. Fornecer à comunidade acadêmica uma valiosa ferramenta para uma perfeita compreensão de um conteúdo tão teórico e de difícil entendimento.

## 3. Metodologia e Funcionamento

O sistema foi criado com base nos estudos e pesquisas realizados por autores consagrados na área, como Tanenbaum (2003), e por Maia et al (2002).

Poderá também ser utilizado principalmente pelo corpo docente das áreas de disciplinas correlatas ao Gerenciamento de Processadores dos mais diversos cursos de nível superior de outras faculdades, como Ciência da Computação, Engenharia da Computação, Sistemas de Informação e semelhantes.

Este simulador utiliza dados fictícios ou reais de processos de um sistema operacional e tem como entrada os dados referentes aos tempos de UCP, que definem quanto tempo o processo permanecerá no sistema operacional desde a sua criação até entrar em estado de término de execução. Na verdade, um processo que possui código de execução definido nunca poderá ter tempo de UCP pré-determinado em um sistema operacional multitarefa, como era possível em sistemas monotarefas, como o MS-DOS®, cujos programas possuíam código medível e curto. Para exemplificar, os códigos *Assembler* permitiam calcular quantos ciclos de processamento eram necessários para se conhecer quando um programa terminaria sua execução. O tempo de UCP é na realidade pós-determinado e pode ser conhecido por meio de um *software* simples. É impossível prever o instante de tempo do seu término devido às diferentes arquiteturas de computadores existentes e à imensa quantidade de código em baixo nível dos programas compilados. Por este motivo, este tempo é introduzido no simulador para emular processos escalonados, a fim de obter dados estatísticos e fazer comparações. Outro dado de entrada é a definição de uma prioridade de execução para cada processo. Com uso dos escalonamentos feitos por prioridades, é possível fazer com que o computador realize mais rapidamente as tarefas mais essenciais ou de missão crítica, como os processos computacionais utilizados em sistemas de telemedicina e embarcados. Outro dado de entrada para este simulador é a atribuição de um *timeslice* ou fatia de tempo, que permite um processador interromper a execução de um processo, colocá-lo no fim de uma fila de processos em estado de pronto e executar outro processo que se encontra no início desta fila. Este dado é importante para este tipo de escalonamento, chamado de escalonamento circular, e evita o problema do *starvation*, isto é, a espera indefinida de um processo por uma oportunidade de ser executado pela UCP.

Todos esses dados de entrada podem ser preenchidos manualmente pelos usuários ou automaticamente pelo sistema simulador para testes com um número bem maior de processos escalonados. Em um sistema operacional servidor, como o *Windows 2003 Server* por exemplo, são executados uma média de 90 processos para uma empresa de pequeno porte que fornece serviços de banco de dados. Neste simulador serão aceitos até, no máximo, 255 processos.

Após a inserção destes dados e a escolha do tipo de escalonamento a ser simulado, poderemos também optar pelo valor de frequência ou ciclos de processamento de uma UCP. Em tempo de execução este parâmetro pode ser modificado.

Quando iniciamos a simulação, observamos os processos saindo do estado de pronto e entrando em estado de execução, uma após o outro, podendo, posteriormente, entrar em estado de pronto por causa do *timeslice*, em estado de espera por causa de requisições feitas pelo processo ao usuário ou ao próprio sistema, ou entrar definitivamente em estado de término. Esta entrada e saída de processos de seus estados é bem visualizada na lista das filas de tipo FIFO do simulador, podendo ser pausada ou colocada em uma frequência baixíssima, para permitir melhor visualização do andamento da simulação.

Durante a simulação também poderemos observar um gráfico de processos em função do tempo, processado em tempo real, que demonstra o uso dos processos pela UCP de acordo com um instante de tempo.

No simulador, poderemos optar por escalonamentos do tipo FIFO, Circulares, por Prioridades e Circular

por Prioridades. Os resultados são bem diferenciados quando utilizamos tipos de escalonamentos diferentes e permite escolhermos o melhor escalonamento para projetos de sistemas operacionais com fins específicos e/ou excepcionais, como aqueles citados anteriormente.

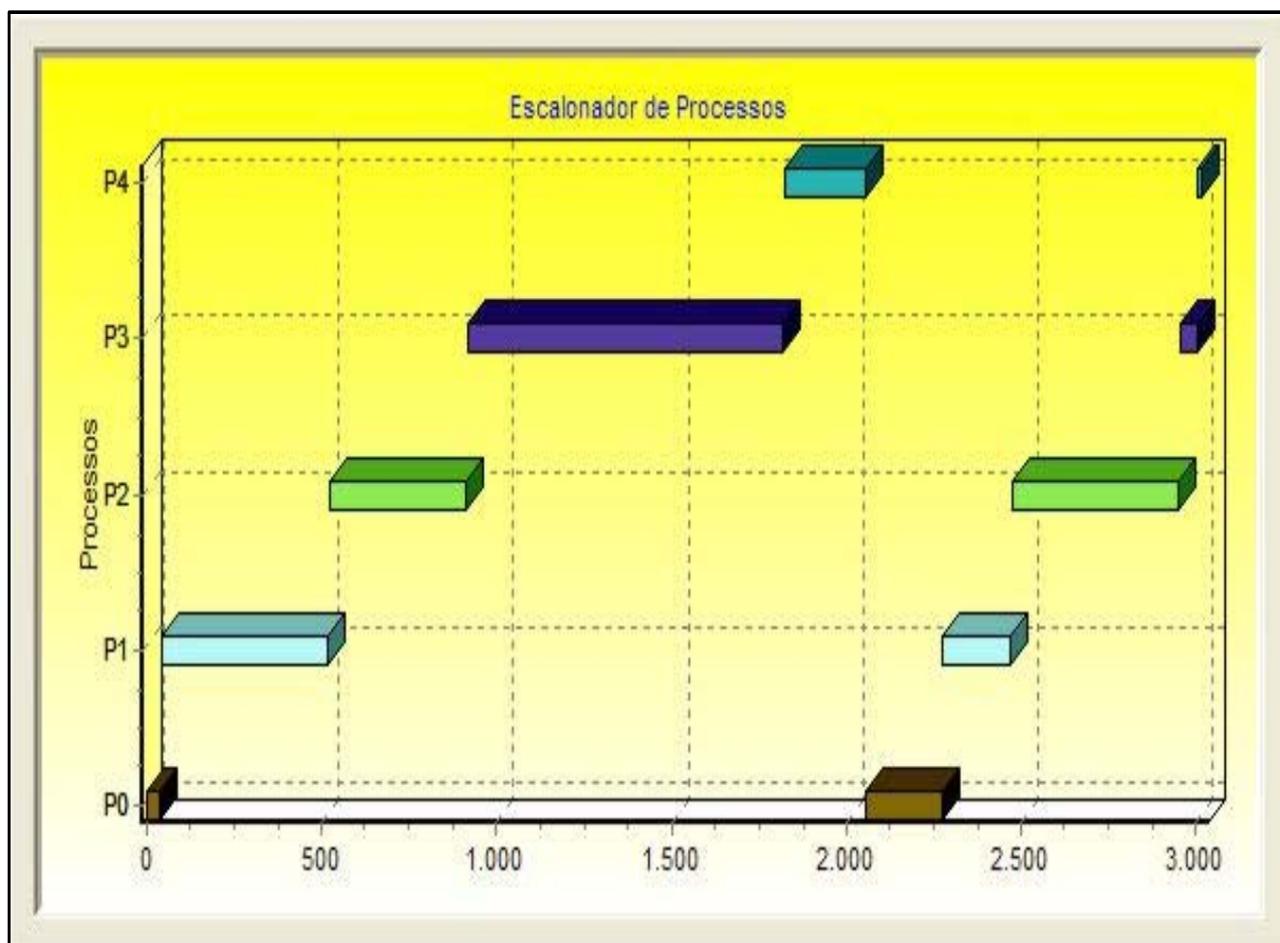
#### 4. Exemplificação

Para uma pequena exemplificação, numa simulação de cinco processos quaisquer criados no instante de tempo 0 (zero) ms, utilizamos como entrada para um escalonamento do tipo FIFO, dados aleatórios, como mostrados na Tab. (1):

**Tabela 1.** Fila de Pronto

Nome do Processo	Tempo UCP pré-determinado
P0	258 ms
P1	688 ms
P2	872 ms
P3	956 ms
P4	250 ms

**Figura (1)** processo x instante de tempo



Numa frequência de 20 Hz, simulamos a concorrência destes processos no uso de um único processador. Esse acesso concorrente traz a vantagem de múltiplos processos serem executados “ao mesmo tempo”. Na Figura (1), podemos ter uma idéia clara da forma como os processos concorrem no uso da UCP em função do instante de tempo decorrido.

Vinte hertz é uma frequência de processamento baixíssima se comparada com os sistemas computacionais atuais, que operam em até 3.2 GHz<sup>2</sup>. Como foi dito anteriormente, devido à alta velocidade destes processadores temos aquela impressão de que todos os aplicativos funcionam ao mesmo tempo. A principal desvantagem da concorrência de processos é que um programa poderá demorar mais a chegar a um estado de término se compararmos os resultados com processos em execução nos sistemas monotarefas.

**Tabela 2.** Estado de Término

Nome do Processo	Tempo de <i>Turnaround</i>
P0	452 ms
P1	775 ms
P2	949 ms
P3	1.000 ms
P4	253 ms
Total: 3.429 ms	

Se compararmos o tempo de UCP da Tab. (1) com o tempo de *turnaround* da Tab. (2) em cada processo, iremos visivelmente enxergar esta diferenciação. Poderíamos utilizar o tempo de UCP da primeira tabela como tempo total de execução para cada processo em um ambiente MS-DOS®, um iniciando a sua execução após o término do outro. Também poderíamos utilizar o tempo de *turnaround* da segunda tabela como se fosse um tempo de UCP para cada processo em um ambiente multitarefa, como os sistemas *Windows NT*® ou o UNIX. Na Tabela (3), vemos o instante de tempo em que o processo terminou definitivamente a sua execução.

**Tabela 3.** Instante em que os processos entraram em estado de término

Nome do Processo	Instante de Tempo de Término
P0	452 ms
P1	1.227 ms
P2	2.176 ms
P3	3.176 ms
P4	3.429 ms
Total: 3.429 ms	

Apesar de o processo P4 ter o menor tempo de UCP, 250 ms, nesta simulação foi o processo que teve o maior atraso para entrar em estado de término, como visto na Tab. (3), devido às requisições feitas pelos outros processos a UCP, de acordo com o critério de ordenação FIFO inicial da fila de criação.

## 5. Conclusão

O presente trabalho versou sobre os escalonamentos de processos em sistemas operacionais e apresentou um projeto concluído de um simulador de escalonamento de processos.

Sugere-se ao corpo docente o uso de simuladores a fim de elucidar conceitos teóricos apresentados em salas de aulas e fazer com que os alunos absorvam com facilidade e interesse a disciplina.

## 6. Referências

MACHADO, F. B; MAIA, L. P. **Arquitetura de Sistemas Operacionais**. 3ª ed. LTC. 2002.

TANEMBAUM, Andrew. **Sistemas Operacionais Modernos**. 2ª ed. Prentice-Hall. 2003

### Responsabilidade de autoria

As informações contidas neste artigo são de inteira responsabilidade de seu autor. As opiniões nele emitidas não representam, necessariamente, pontos de vista da Instituição e/ou do Conselho Editorial.

<sup>2</sup> Até o desenvolvimento deste artigo em janeiro de 2006.