

Trabalho de Sistemas Operacionais

Professor: Anderson Valadão

Alunos: Matheus Soares Tostes Miguel Eduardo da Silva

Relatório – Simulação de Gerenciamento de Processos e Memória

Gerenciamento de Memória

Nesta etapa, foi realizada a simulação de diferentes técnicas de alocação de memória, abrangendo os algoritmos de particionamento contíguo — First Fit, Best Fit e Worst Fit — e o método de paginação. Essas abordagens permitiram observar como a forma de alocar espaço para processos impacta a fragmentação da memória e o desempenho geral do sistema.

1. Particionamento Contíguo:

O particionamento contíguo simula uma memória dividida em blocos contínuos, onde os processos são alocados de acordo com diferentes estratégias:

First Fit: aloca o processo na primeira partição livre que comporta seu tamanho. Apresentou bons resultados para processos menores posicionados antes na lista, porém, causou fragmentação quando blocos maiores foram parcialmente utilizados.

Best Fit: escolhe a partição com menor sobra de espaço. Teve melhor aproveitamento da memória, mas exigiu mais processamento por necessitar percorrer todas as partições para encontrar o melhor encaixe. Também pode criar partições pequenas demais para uso futuro.

Worst Fit: seleciona a maior partição livre disponível. Embora tente minimizar a fragmentação ao dividir grandes blocos, pode deixar grandes áreas da memória mal aproveitadas se o processo alocado for muito pequeno.

Em todos os casos, foi necessário controlar o número de partições manualmente e realizar divisões para representar sobras de memória. A função `imprimir_mapa_memoria()` foi essencial para visualizar o resultado final da alocação.

2. Paginação:

Na simulação de paginação, a memória foi dividida em quadros fixos e os processos em páginas. Isso elimina a fragmentação externa e facilita a alocação, permitindo que páginas de diferentes processos coexistam em quadros distintos.

Ao alocar páginas para três processos diferentes (com 15, 8 e 20 páginas), foi possível observar como a tabela de quadros se preenche progressivamente. Se a memória não possuir quadros suficientes, o processo é parcialmente alocado, e a simulação exibe uma mensagem informando a quantidade de páginas efetivamente atribuídas.

Este método se mostrou mais eficiente e organizado para cenários com múltiplos processos de tamanhos variados, além de ser mais flexível que o particionamento contíguo.

Gerenciamento de Processos

Durante a implementação dos algoritmos de escalonamento de processos — FCFS (First Come First Serve), SJF (Shortest Job First) e Round Robin — foi necessário estruturar uma fila robusta que suportasse a manipulação dinâmica dos processos. A estrutura FILA precisou garantir integridade nos métodos de enfileiramento, desenfileiramento e reinserção (no caso do Round Robin).

No algoritmo SJF, a necessidade de ordenar os processos pelo tempo restante trouxe o desafio adicional de implementar uma ordenação interna (Bubble Sort), o que funcionou bem para a baixa escala da simulação, mas poderia ser ineficiente em sistemas maiores.

Já no Round Robin, o cuidado com o tempo de execução (quantum) exigiu controle preciso da reinserção dos processos na fila e decremento de tempo restante, evitando loops infinitos e garantindo que o processo fosse finalizado corretamente.

Eficiência das Políticas de Escalonamento

Round Robin foi o mais eficiente para múltiplos processos com tempos variados, por dividir o tempo da CPU de forma justa entre todos, sendo ideal para sistemas multitarefa e interativos.

FCFS se destacou pela simplicidade, mas sofreu com o efeito de espera longa, prejudicando a experiência em situações em que o primeiro processo era muito demorado.

SJF ofereceu o melhor tempo médio de espera, porém introduziu risco de starvation, já que processos com tempos longos podem nunca ser executados se sempre houver processos mais curtos.

Impacto do Gerenciamento de Memória no Desempenho

O desempenho do escalonamento está diretamente ligado à forma como a memória é gerenciada. A falta de espaço ou fragmentação pode impedir a entrada de novos processos, afetando a fila de escalonamento. O particionamento contíguo, apesar de simples, mostrou-se mais suscetível à fragmentação. Já a paginação se destacou por sua capacidade de otimizar o uso da memória, permitindo o carregamento parcial de processos e eliminando a necessidade de blocos contíguos.

Além disso, algoritmos como Round Robin, que realocam processos constantemente, beneficiam-se de um sistema de memória mais dinâmico e flexível como o da paginação.

Conclusão Final

A combinação da simulação de políticas de escalonamento e técnicas de gerenciamento de memória demonstrou na prática os desafios e decisões que um sistema operacional precisa tomar. A escolha da técnica mais adequada depende do perfil da aplicação e da carga de trabalho do sistema.

Para ambientes de tempo real e interação, Round Robin com paginação oferece a melhor responsividade. Para cargas previsíveis com foco em tempo de espera, SJF com particionamento bem gerenciado pode ser mais eficiente.

Para sistemas simples, FCFS e First Fit ainda são soluções válidas.

Essas simulações reforçam como memória e processos são áreas integradas dentro de um sistema operacional, e seu gerenciamento conjunto é essencial para um desempenho equilibrado, justo e eficiente.