

Simulador de Memória Virtual

Bárbara, Eduarda, Juliana

Sistemas Operacionais 2018/1

1 Resumo do projeto

Este trabalho apresenta um simulador de memória virtual. O simulador recebe como entrada um arquivo que contém a sequência de endereços de memória acessados por um programa. Esses endereços estão escritos como números hexadecimais, seguidos por uma letra R ou W, para indicar se o acesso foi de leitura ou escrita. Ao iniciar o programa, será definido o tamanho da memória (em quadros) para aquele programa e qual o algoritmo de substituição de páginas a ser utilizado. O programa deve, então, processar cada acesso à memória para atualizar os bits de controle de cada quadro, detectar faltas de páginas (page faults) e simular o processo de carga e substituição de páginas, por meio de um dos três algoritmos implementados (Segunda chance, LRU ou NRU).

Para simular a memória virtual, foi implementada uma tabela de páginas reversa utilizando hash. Para cada página, calcula-se o seu endereço na tabela, que determina em qual item ela estará. Se já possui outra página neste endereço, faz-se essa página apontar para a nova, formando então uma lista encadeada.

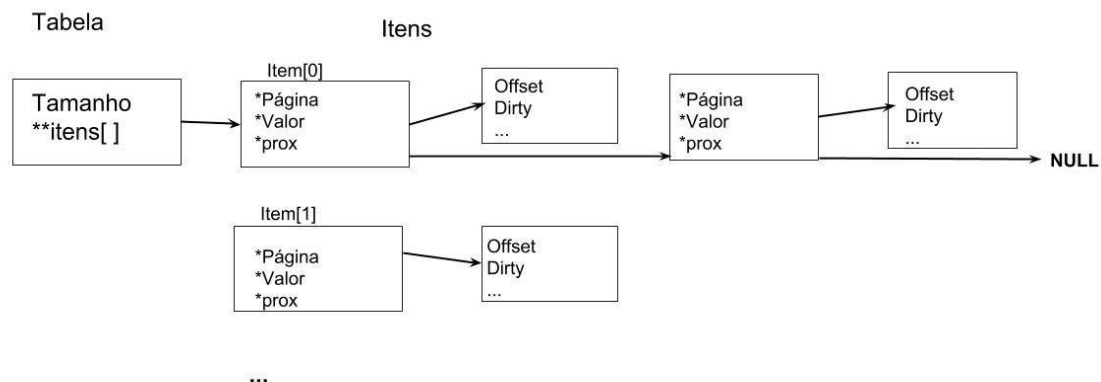


Figura 1 - Tabela de páginas invertida

Para que não seja necessário percorrer toda a tabela em busca da página a ser removida, foi utilizada uma lista como estrutura auxiliar. Desse modo, as páginas podem ser adicionadas e removidas da lista em uma ordem que facilita sua identificação na hora de fazer a substituição. Toda página adicionada à tabela é adicionada também ao final da lista.

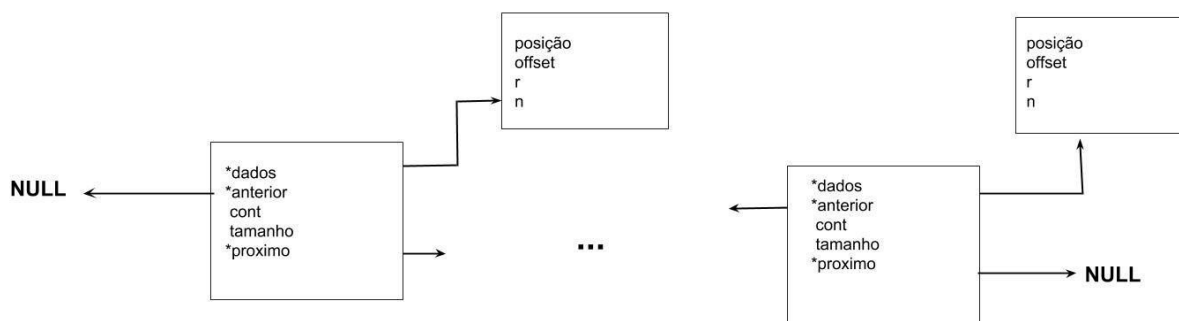


Figura 2 - Lista de páginas

Ao retirar uma página da tabela, a nova página não necessariamente ocupará a mesma moldura, pois seu endereço deve ser calculado de acordo com sua hash. Na lista guarda-se a posição e o offset daquela página, portanto para encontrá-la na tabela de páginas, deve ir até aquela posição e comparar todos os offsets que nela estão.

1.1 Algoritmos de substituição

- Segunda chance: Toda página é adicionada ao final da lista, logo as carregadas a mais tempo estão no início da lista. A página a ser retirada está no início da lista, porém se seu bit R é igual a 0, ela é removida da primeira posição e adicionada ao final da lista, tendo seu bit R modificado para 1.
- LRU: Toda vez que a página é acessada, ela é realocada para o final da lista, logo as páginas menos recentemente acessadas serão as primeiras. Portanto, a primeira posição será removida e a nova página é inserida ao final.
- NRU: Toda página de escrita tem seu bit M igual a 1, e toda vez que uma página é referenciada, seu bit R é colocado como 1. Sendo assim, procura-se

na lista a página a ser substituída, sendo que a ordem de substituição de páginas é seguinte:

- 1 - $R = 0$ e $M = 0$;
- 2 - $R = 0$ e $M = 1$;
- 3 - $R = 1$ e $M = 0$;
- 4 - $R = 1$ e $M = 1$;

2 Análise de desempenho

As Figuras 3 e 4 abaixo apresentam a falta de página média para cada algoritmo. Foram realizados testes variando o número de páginas e tamanho da memória para cada um.

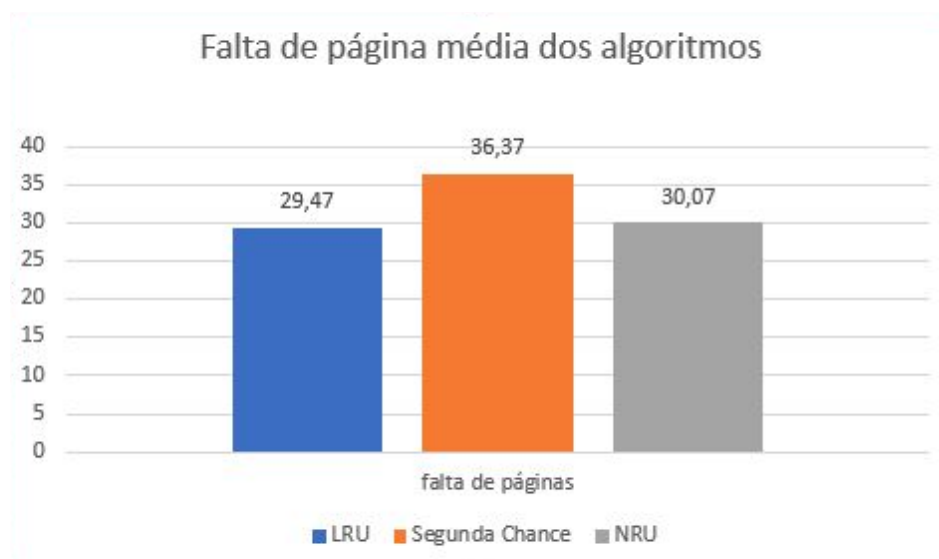


Figura 3 - Falta de página média dos algoritmos

De acordo com a Figura 3, percebe-se que o algoritmo Segunda Chance apresentou maior número de falta de páginas em relação aos demais. O algoritmo LRU teve o melhor desempenho, com resultados bem próximos ao algoritmo NRU.

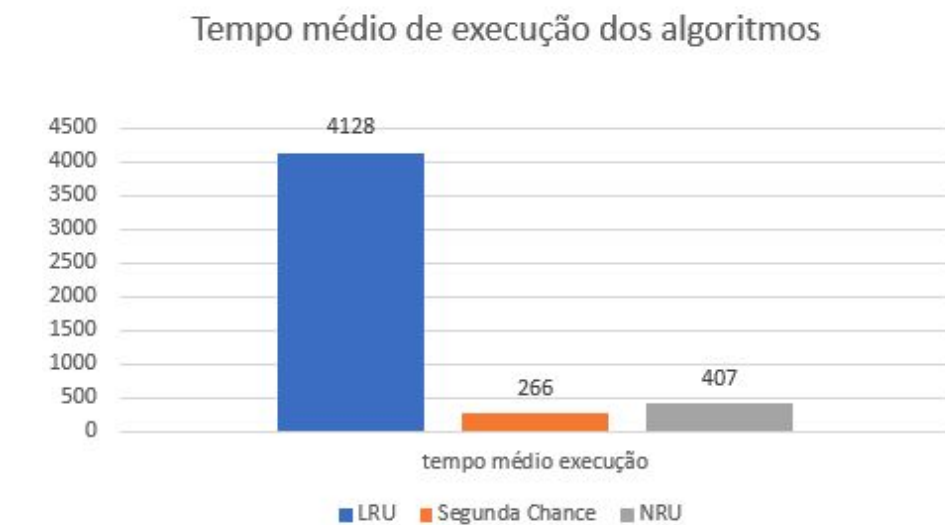


Figura 4 - Tempo médio de execução dos algoritmos em ms

Já na Figura 4 observa-se o tempo médio de execução dos algoritmos. Em geral, os tempos foram bem próximos, com exceção do algoritmo LRU, que ao aumentar consideravelmente o tamanho da memória e número de páginas, teve sua execução na casa dos segundos ou minutos.

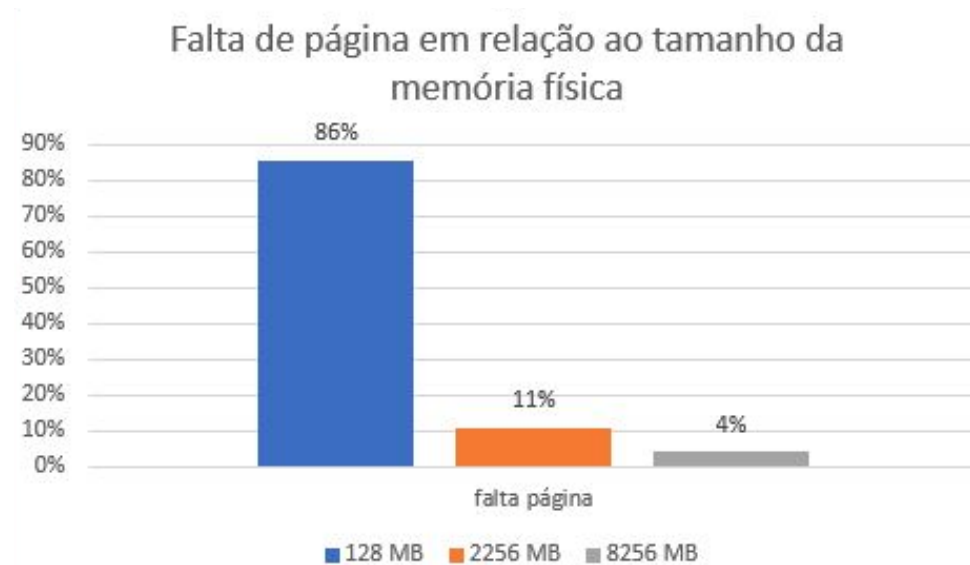


Figura 5 - Falta de página em relação ao tamanho da memória física

A Figura 5 apresenta a porcentagem média de falta de páginas ao se variar a memória física. Para obter os resultados, os testes foram feitos fixando o tamanho da página em 8K, e executando para cada algoritmo. A partir dela, percebe-se que quanto maior a memória física disponível, menor o número de faltas de página.

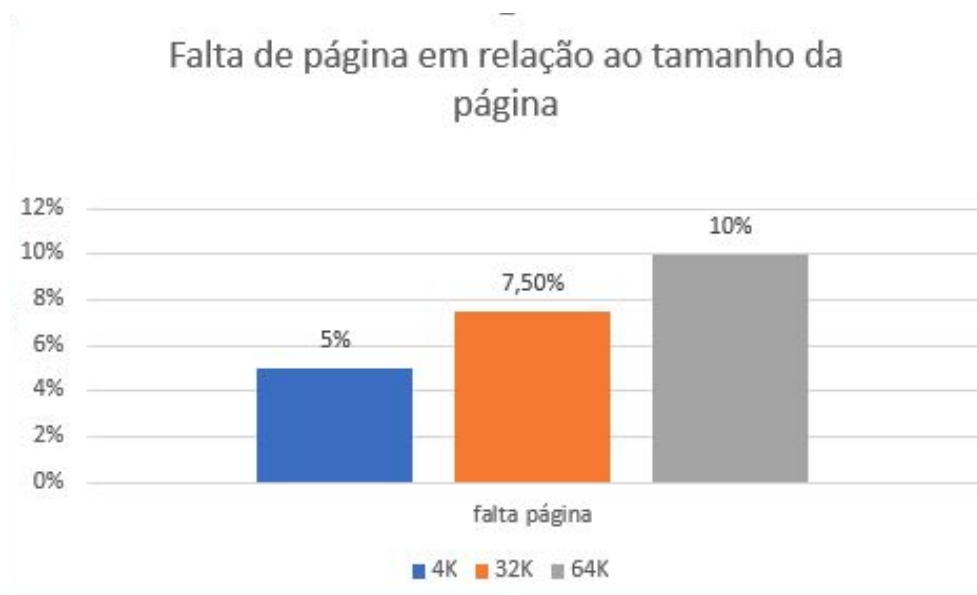


Figura 6 - Falta de página em relação ao tamanho da página

Por fim, a Figura 6 mostra o comparativo do número de falta de páginas ao variar o tamanho da página. Para isso, fixou-se o tamanho da memória física em 2256 MB e executou o programa para todos os algoritmos. Observa-se que ao aumentar o tamanho da página, também aumenta-se o número médio de falta de página.