Hugo de Barros Pimentel Dutra Machado - 1410530

Newton Coelho – 1621351

**Relatório de Desempenho :**

No nosso projeto , cada endereço virtual dos arquivos .log foram divididos da seguinte forma : O índice(página da tabela) é composto pelos 12 bits mais significativos do endereço virtual , que compõe 4096 Kb , e os 20 bits restantes pertencendo ao offset , compondo 2^20 Kb.

Com a função de leitura dos arquivos passada nas especificações do trabalho, os arquivos são lidos linha a linha, assim pegamos o endereço e separamos em offset e índice. A cada linha lida inicialmente, pega o índice do endereço virtual e na função ProcuraTabela, que está dentro da função "trans", é retornado a página correspondente ao índice do endereço virtual, sendo que dos 4096 bits de páginas, apenas 256 são mapeados. Durante a leitura dos novos endereços e pegando os índices de cada um, é feita uma verificação se esse índice está mapeado na tabela de páginas. Caso esteja mapeado, ele apenas imprime o endereço lido e aumenta a frequencia em 1. Caso o valor retornado do índice seja -1, quer dizer que ele não esta mapeado para a memória física , e caso o índice lido seja diferente dos 256 mapeados, ocorre um Page fault, ele realiza o algoritmo LFU de substituição de página , buscando o de menor freqüência , mudando a página de tabela para -1 do processo que perdeu, e o processo que ganhou o frame recebe o índice do frame.

No gerenciador de memória , para saber quem tem que fazer a troca de páginas através do SIGACTION , foi usado uma função para pegar o pid do processo que está enviando o sinal(Este sinal é enviado no trans). O array pidFilho possui os pids de cada processo, assim conseguimos saber em qual processo ocorreu o page fault. Assim, damos um SIGSTOP nesse processo e logo em seguida no GM é feita a troca de páginas na função "TrocaPagina" , realizando o algoritmo LFU, e no final é dado um SIGCONT no pidFilho que parou.

Análise do Desempenho :

Os primeiros 256 page faults relacionados ao mapeamento inicial da Tabela de Páginas não estão sendo contados por serem triviais no processo( Ocorrerem sempre da mesma forma).

Inicializamos os processos de forma igual , mas na hora de criar a page table o computador acaba colocando todo o mapeando inicial para um processo só, nos 256 mapeamentos iniciais.

Quando o mapeamento está cheio , o algoritmo de substituição começa a ser realizado e os page faults tendem a ocorrer de uma maneira mais equilibrada . Também no início o número de escritas no HD tendem a ser mínimos , ou seja , ocorrem com pouca freqüência , pois o algoritmo de substituição prioriza os endereços R(Read) , ocorrendo menos trocas de páginas com endereços W(Write).

Essa análise foi observada com base no arquivo desempenho.log, gerado na execução do programa.