Universidade Federal do Pará
Instituto de Tecnologia – ITEC
Faculdade de Computação e Telecomunicações
Disciplina: Sistemas Operacionais
Docente: Diego Lisboa Cardoso
Discentes: Adryele Costa de Oliveira - 202206840035
Amanda Gabrielly Prestes Lopes - 202207040043
Felipe Daniel Setubal Alves - 202306840011
Filipe Corrêa da Silva - 202006840020
Lívia Stefane Hipólito Pires – 202206840043

Memória Swap

Belém - Pará
2023

1. Testes Memória Swap

Para o teste com a memória swap foi utilizado o código a seguir, que aloca dinamicamente uma matriz de 10000x10000.

```c
#include <stdio.h>
#include <stdlib.h>

int main() {
    // Tamanho da matriz
    int linhas = 10000;
    int colunas = 10000;
    // Alocação dinâmica da matriz
    int** matriz = (int*)malloc(linhas * sizeof(int*));
    if (matriz == NULL) {
        fprintf(stderr, "Erro ao alocar memória para as linhas da matriz.\n");
        return 1;
    }
    for (int i = 0; i < linhas; i++) {
        matriz[i] = (int*)malloc(colunas * sizeof(int));
        for(int j = 0; j < colunas; j++) {
            matriz[i][j] = 0;
        }
        if (matriz[i] == NULL) {
            fprintf(stderr, "Erro ao alocar memória para as colunas da
matriz.\n");
            // Libera as linhas já alocadas
            for (int j = 0; j < i; j++) {
                free(matriz[j]);
            }
            // Libera o array de linhas
            free(matriz);
            return 1;
        }
    }
    for (int i = 0; i < linhas; i++) {
        for(int j = 0; j < colunas; j++){
            matriz[i][j] = matriz[j][i];
        }
    }
    // Liberação da memória alocada
    for (int i = 0; i < linhas; ++i) {
        free(matriz[i]);
    }
    free(matriz);
    return 0;
}
```

Foram conduzidos testes abordando o desempenho da memória swap em uma máquina virtual hospedada no VirtualBox, utilizando o sistema operacional Linux Mint. O experimento consistiu em seis testes distintos, sendo os três primeiros executados em uma configuração de máquina virtual com 2 GB de RAM, seguidos pelos três últimos testes realizados com uma configuração de 1 GB de RAM.

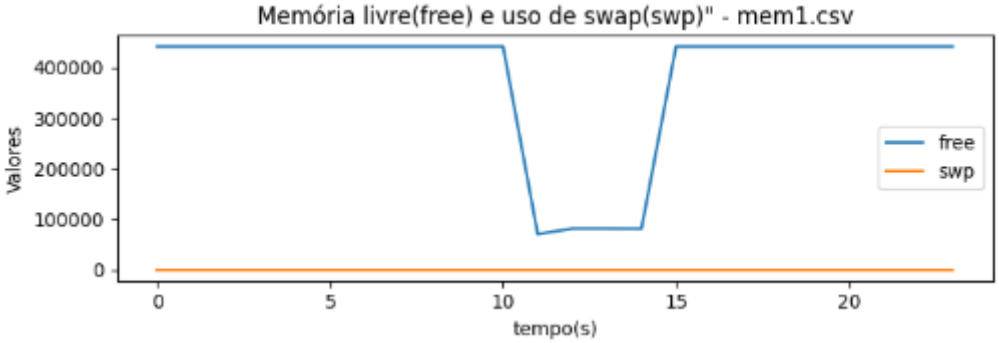
Após a conclusão de cada teste, a memória de swap foi resetada, desativada e reativada. Essa abordagem permitiu uma análise mais aprofundada do comportamento do sistema em relação à memória swap.

Os testes serão abordados como mem1, mem2, mem3, mem4, mem5 e mem6 nos gráficos, para a análise sobre os impactos foram usados os comandos *`htop`*, para visualizar o consumo de memória, *`time`*, para saber o tempo de execução do processo, e *`vmstat -t 1`*, para obter as informações para plotagem dos gráficos.

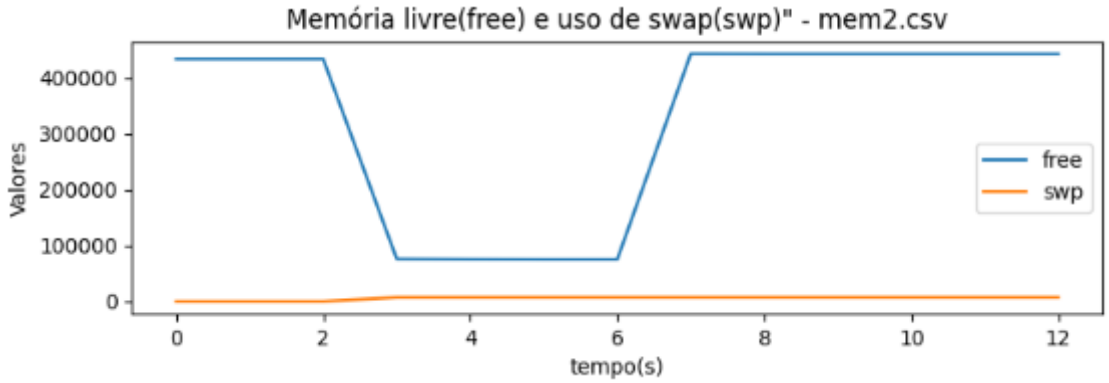
2. Testes

2.1. Testes com 2GB de memória

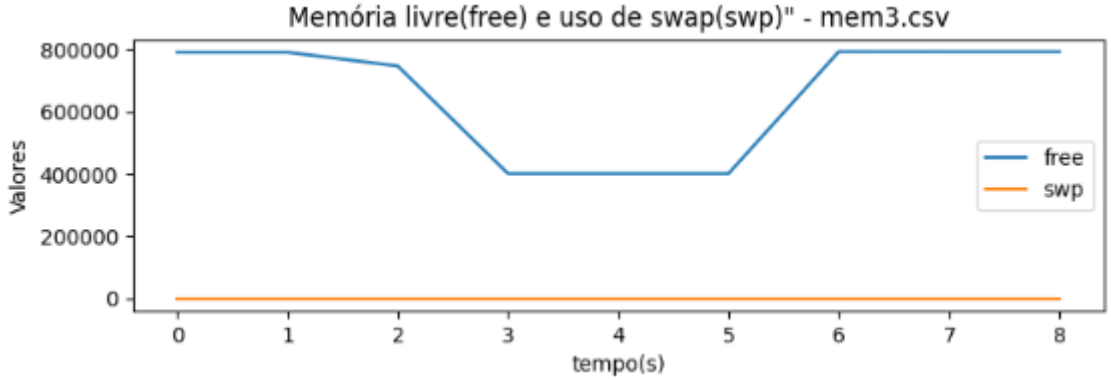
O teste mem1 foi conduzido sem a utilização da memória swap, resultando em uma média de tempo de execução de 4.16 segundos. No início do teste, a utilização de memória era de 1.22 GB, atingindo um pico de 1.59 GB durante a execução. O gráfico a seguir ilustra a quantidade de memória livre ao longo da execução do programa.



No teste mem2, foi alocada uma memória swap de 1 GB, resultando em um tempo médio de execução de 4.02 segundos. A utilização de memória no início e durante o teste foi consistente com o teste anterior, mantendo-se em 1.22 GB no início e atingindo 1.59 GB durante a execução. O gráfico a seguir apresenta a dinâmica da utilização da memória swap e a quantidade de memória livre ao longo do teste.



No teste mem3, o último com 2 GB de memória, o total de swap disponível foi aumentado para 3 GB. Apesar da consistência no consumo de memória no início e durante o teste, houve uma ligeira redução no tempo médio de execução, alcançando 3.84 segundos, que pode não ter relação com o aumento de swap, já que a mesma foi subutilizada. O gráfico a seguir representa a dinâmica da execução durante este teste, incluindo a utilização da memória swap e a quantidade de memória livre.



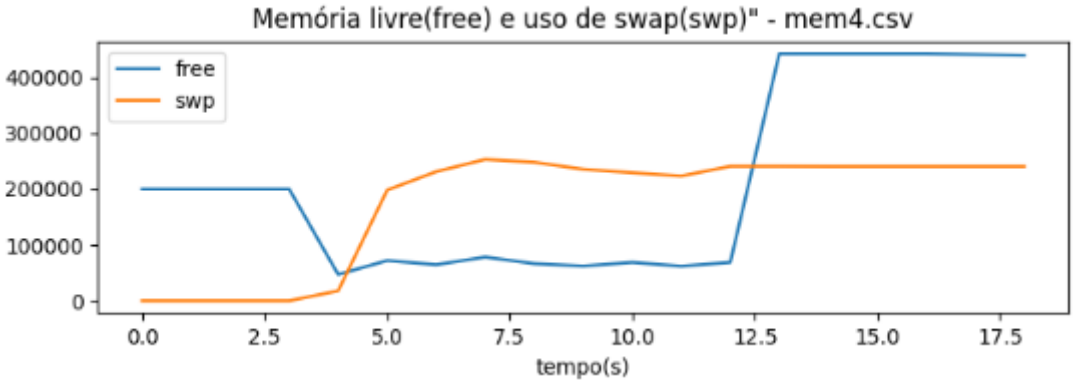
Durante os três testes realizados, observou-se uma variação mínima no tempo de execução do código e na utilização da memória principal e de swap. Essa estabilidade pode ser atribuída à disponibilidade adequada de memória no sistema para a execução do programa, não criando a necessidade da utilização do swap.

2.1. Testes com 2GB de memória

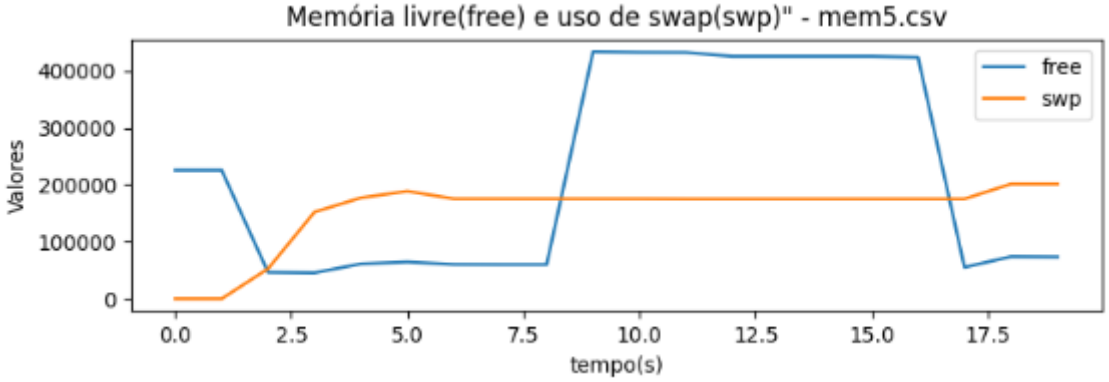
Nos testes subsequentes, com apenas 1 GB de memória RAM, as execuções utilizaram, respectivamente, 1 GB, 3 GB e 0 GB de memória swap. No primeiro teste com essa quantidade de RAM (mem4), a primeira execução do programa apresentou um tempo de execução superior a 3 minutos. Ao realizar execuções subsequentes, essa primeira execução foi descartada devido à sua discrepância em relação aos outros dados obtidos. Isso foi feito para garantir uma melhor análise dos dados.

Nas execuções válidas do teste 4, a quantidade de memória utilizada antes da primeira execução foi de 632 MB. Durante a execução, o consumo de memória atingiu o limite da RAM e, após a execução, o consumo estabilizou em 472 MB. Esses acontecimentos levam a conclusões que serão discutidas posteriormente.

A média de tempo de execução para a primeira execução do programa foi de 14.5 segundos. No entanto, para as execuções seguintes, essa média caiu significativamente, para 3.76 segundos. Nesse teste foi perceptível o uso do swap. O gráfico a seguir representa a dinâmica da memória durante a primeira execução do programa neste teste.



No 5º teste, o total de swap disponível foi de 3 GB. Antes do teste, a quantidade de memória consumida era de 635 MB, e após a execução, o consumo caiu para 460 MB. A média de tempo para a primeira execução foi de 11 segundos, e para as execuções subsequentes, a média caiu para 4 segundos. O gráfico abaixo representa a dinâmica da memória durante a primeira execução do programa neste teste.



No teste 6, não foi possível coletar dados devido à escassez de memória disponível. O computador travou, e o sistema operacional encerrou o processo. Essa situação indica que a execução do programa não é viável com apenas 1 GB de RAM.

Nos últimos três testes, a utilização significativa da memória swap foi evidente devido à escassez de memória disponível. Além disso, foi observada uma redução notável no tempo de execução após a primeira execução. Isso ocorreu porque parte dos processos já estava armazenada na memória swap, eliminando a necessidade de realocar os arquivos da memória principal para a memória de swap durante essas execuções subsequentes. Nos gráficos de mem4 e mem5 é possível perceber que o que foi armazenado em swap não foi transferido devolta para a memoria principal mesmo que houvesse espaço livre, isso ocorre pois não havia necessidade de utilizar o que foi transferido.

3. Conclusão

A utilização da memória swap possibilita a execução de programas mais pesados ou de um maior número de programas, mesmo quando não há espaço suficiente para alocar tudo na memória principal. Além disso, observa-se uma pequena variação no uso da swap entre os testes com 1 e 3 GB disponíveis, indicando que grandes quantidades de swap podem não necessariamente resultar em um aumento significativo de desempenho.

É importante ressaltar que, embora a memória swap possa proporcionar uma eficiência semelhante em alguns casos, ela não possui a mesma eficácia da memória RAM. No entanto, ela representa uma solução simples e econômica para melhorar a eficiência do sistema, permitindo a execução de tarefas mais pesadas ou a gestão de múltiplos programas, quando a capacidade da memória principal é limitada.