



RELATÓRIO PROJETO 2

Simulador de Paginação de Memória **Sistemas Operacionais**

GRUPO

Bruna Soncini RA:10428267

Julia Andrade RA:10427828

Gabriel Buck RA:10425384

Henrique Catanha RA:10426708

Lucas Akio RA:10425346



1. Introdução

Este projeto tem como objetivo desenvolver um simulador de paginação que traduza endereços virtuais para endereços físicos, utilizando os algoritmos de substituição de páginas FIFO (First-In, First-Out) e Random.

O simulador suporta múltiplos processos e traz estatísticas de desempenho, como taxa de page faults e hits, possibilitando a análise do comportamento dos algoritmos implementados.

2. Descrição da Implementação

O simulador é composto pelas seguintes estruturas de dados principais:

- **Simulador:** mantém informações globais como tamanho da página, tamanho da memória física, contadores de acesso, número de processos, escolha do algoritmo e estado da memória física.
- **Processo:** contém o identificador do processo (PID), tamanho do espaço de endereçamento, número de páginas e a tabela de páginas.
- **Página:** representa cada página do processo, indicando se está presente na memória física, o frame em que está carregada e o tempo em que foi carregada para gerenciamento de substituição.
- **Memória Física:** vetor de frames que armazenam as páginas atualmente carregadas e tempos de carga para auxiliar nos algoritmos de substituição.

Algoritmos Implementados

- **FIFO (First-In, First-Out):** substitui a página que está na memória há mais tempo, identificada pelo menor tempo de carga.
- **Random:** escolhe aleatoriamente um frame para substituir.

Fluxo do Programa

O programa inicializa os processos com tabelas de páginas zeradas e a memória física vazia. Através de um menu, o usuário seleciona o algoritmo de substituição. Durante a simulação, o programa recebe uma sequência de acessos de endereços virtuais para cada processo. Cada endereço é traduzido



para a página e deslocamento. Se a página não está presente, ocorre um page fault e a página é carregada na memória usando o algoritmo selecionado. Caso contrário, contabiliza-se um hit. Após cada acesso, o estado da memória física é exibido nas estatísticas.

3. Análise Comparativa dos Algoritmos

Para o algoritmo FIFO, observou-se que os page faults ocorrem inicialmente, quando as páginas ainda não estão carregadas. Com o progresso da simulação, algumas páginas permanecem na memória, gerando hits e diminuindo a taxa de page faults. A saída exibiu o estado atualizado da memória física após cada acesso, evidenciando a substituição de páginas conforme a política escolhida. As estatísticas finais mostraram o total de acessos, hits, page faults e suas respectivas taxas, permitindo análise do desempenho do algoritmo.

4. Conclusão

O projeto alcançou seu objetivo ao implementar um simulador funcional de paginação, com suporte a múltiplos processos e algoritmos FIFO e Random.

A simulação permitiu compreender o impacto dos algoritmos na ocorrência de page faults e hits, reforçando a importância da escolha adequada de política de substituição para otimização do desempenho do sistema.

Como melhorias futuras, recomenda-se a implementação de algoritmos adicionais, como LRU e Clock.

5. Referências

- <https://sdpm-simulator.netlify.app/>
- https://github.com/romulo-martins/cachesim-siscomp-uff?utm_source=chatgpt.com
- <https://youtu.be/kJBpG3v2P5o?si=11vhGzFuUQeoHn3p>