



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
DEPARTAMENTO DE ENGENHARIAS E TECNOLOGIA - DETEC
CAMPUS PAU DOS FERROS
CURSO DE TECNOLOGIA DA INFORMAÇÃO

TRABALHO REFERENTE À 3ª UNIDADE - PEX0093 - SISTEMAS OPERACIONAIS (1200276) - TURMA 01

1. Implementação do algoritmo de substituição de páginas sorteado e elaboração do relatório técnico

A atividade consiste na implementação do algoritmo sorteado em sala: https://docs.google.com/spreadsheets/d/1R0rXogp_oGESTQJDQqbmbzRiglHWxUGBZGCcTa6YLdo/edit?usp=sharing. O slide base sobre esse trabalho pode ser encontrado no link: <https://github.com/rai-e/Aulas-sistemas-operacionais/tree/main/3%20unidade/trabalho%20pr%C3%A1tico> (Slide trabalho prático 3 unidade.pdf). Cada grupo deve estudar e compreender o funcionamento do algoritmo designado, garantindo que a implementação esteja correta e eficiente.

Por questões de simplicidade, o programa começará com a RAM e o HD já cheios, com a seguinte configuração:

```
/* Considere que o sistema está no seguinte estado
* - A é composto de 3 blocos/páginas
*   PARTE 1 DE A NO DISCO
*   PARTE 2 DE A NO DISCO
*   PARTE 3 DE A NA RAM
* - B é composto de 4 blocos/páginas
*   PARTE 1 DE B NO DISCO
*   PARTE 2 DE B NO DISCO
*   PARTE 3 DE B NO DISCO
*   PARTE 4 DE B NO DISCO
* - C é composto de 3 blocos/páginas
*   PARTE 1 DE C NA RAM
*   PARTE 2 DE C NA RAM
*   PARTE 3 DE C NA RAM
*   [ 0 , 1 , 2 , 3 , 4 , 5 ]
* RAM [PARTE 3 DE A, PARTE 1 DE C, PARTE 2 DE C, PARTE 3 DE C]
* DISCO [PARTE 1 DE A, PARTE 2 DE A, PARTE 1 DE B, PARTE 2 DE B, PARTE 3 DE B, PARTE 4 DE B]
* */
```

```

/* preenchendo a ram e disco com suas páginas cada um */
/* CRIANDO PÁGINAS DE A*/
hd[0] = new Page("PARTE 1 DE A");
hd[1] = new Page("PARTE 2 DE A");
ram[0] = new Page("PARTE 3 DE A");
/* CRIANDO PÁGINAS DE B*/
hd[2] = new Page("PARTE 1 DE B");
hd[3] = new Page("PARTE 2 DE B");
hd[4] = new Page("PARTE 3 DE B");
hd[5] = new Page("PARTE 4 DE B");
/* CRIANDO PÁGINAS DE C*/
ram[1] = new Page("PARTE 1 DE C");
ram[2] = new Page("PARTE 2 DE C");
ram[3] = new Page("PARTE 3 DE C");

```

O programa consiste em um menu interativo com 4 opções:

1 - REALIZAR SUBSTITUIÇÃO DE PÁGINA

- Solicita qual índice i do vetor hd que deve sofrer o page in
- Exibe qual página está nesse índice i do hd
- Seu algoritmo de substituição de páginas deve selecionar algum índice j do vetor/LinkedList para sofrer o page out
- Exibe qual página está nesse índice j da RAM
- Swap de variáveis: $ram[j] \longleftrightarrow hd[i]$

2 - REFERENCIAR QUADRO DA RAM

- Solicita qual índice j do vetor/LinkedList ram que deve ser referenciado
- Exibe qual página está nesse índice j da ram
- Algum processamento adicional que seu algoritmo necessite

3 - VISUALIZAR RAM E DISCO ATUALMENTE

- Exibe o vetor/LinkedList ram
- Exibe o vetor hd
- Obs.: Caso a classe da Página possua outros atributos, todos devem ser exibidos
 - Page{ String dados; int cont; long tempo; double x}
 - dados, cont, tempo e x devem aparecer no "print"

0 - SAIR

- Encerra o programa

A classe página consiste de uma String simples para representar o conteúdo da página. Obs.: alguns algoritmos podem necessitar de mais atributos.

O arquivo base desse programa em Java é encontrado no link a seguir: <https://github.com/rai-e/Aulas-sistemas-operacionais/tree/main/3%20unidade/trabalho%20pr%C3%A1tico>. Os arquivos base são GerenciadorMemoria.java e Page.java.

Após a implementação, cada grupo deve elaborar um relatório técnico descrevendo todo o processo. O relatório deve conter os seguintes elementos:

Introdução:

- A contextualização do problema de substituição de páginas;
- Objetivo geral do relatório;
- Breve explicação de funcionamento do algoritmo sorteado (slide [Slide Trabalho prático 3 unidade.pdf](#) - páginas 7 a 16) ;
- As vantagens e desvantagens do algoritmo implementado (slide [Slide Trabalho prático 3 unidade.pdf](#) - páginas 7 a 16).

Desenvolvimento:

- IDE (integrated development environment) utilizada;
 - VSCode, Visual Studio, Eclipse, Netbeans
- Linguagem utilizada;
 - Java, Javascript, Python, C, C++, entre outros
- Estruturas de dados utilizadas e o que elas representam no algoritmo;
 - Array, fila, pilha, lista, árvore, tabela de dispersão, entre outros
- Explicação detalhada sobre como o algoritmo foi implementado;
- Citar qual a política de busca de páginas usada (slide [Slide Trabalho prático 3 unidade.pdf](#) - página 6). Não é necessário aprofundamento nesse ponto.

Conclusão:

- Considerações finais sobre a implementação

Esse documento deve conter capa (exemplo no final deste documento), introdução, desenvolvimento e conclusão. Obs.: Podem ser adicionadas outras seções ou subseções a critério dos integrantes.

2. Links relacionados

- Link do Discord: <https://discord.gg/njT8K8cf>.
- Link do Github:
<https://github.com/rai-e/Aulas-sistemas-operacionais/tree/main/3%20unidade/trabalho%20pr%C3%A1tico>.
 - EpochExemplo.java - Uso de timestamp em Java
 - FilaExemplo.java - Uso de Queue e LinkedList como uma fila
 - NumerosAleatorios.java - Uso do gerador de números pseudoaleatórios
 - GerenciadorMemoria.java - Classe base da implementação
 - Page.java - Classe referente à pagina da implementação
 - Slide trabalho prático 3 unidade.pdf - Slide com explicação sobre os algoritmos e o relatório
 - Trabalho prático 3 unidade.pdf - Descrição do trabalho prático da 3ª unidade

3. Envio dos relatórios

O relatório pode conter o link para o GitHub, Google Drive, OneDrive ou outros repositórios de arquivos *online* com o código do algoritmo. Como alternativa, os arquivos-fonte podem ser anexados ao e-mail de envio do relatório.

Os grupos devem ser formados com, no máximo, 5 ou 6 pessoas. Um único relatório deve ser gerado em .pdf e um único envio por grupo é necessário.

O envio será via e-mail para raiemmanuel50.50@gmail.com.

O prazo da entrega é até as 23h59min59s do dia 15/02/2025. (Podemos falar sobre prorrogação, se necessário)

O assunto do e-mail deve ser: Trabalho 3 unidade - [Nome do líder do grupo] - PEX0093 - SISTEMAS OPERACIONAIS (1200276) - Turma 01

4. Bibliografia sugerida

TANENBAUM, A. S. Sistemas Operacionais modernos. 3ª ed. São Paulo: Prentice Hall, 2010. 672p.

DEITEL, H.; DEITEL, P.; STEINBUHLER, K. Sistemas Operacionais. 3ª ed. São Paulo: Prentice Hall, 2005. 784p.

MACHADO, F. B.; MAIA, L. P. Arquitetura de Sistemas Operacionais. 5ª ed. Rio de Janeiro: LTC, 2013. 266p.



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO XYZ
CURSO DE XYZ

Nome do(a) autor(a)
(Apenas iniciais em maiúsculo)

Título: Subtítulo
(Iniciais e nomes próprios ou científicos devem ser em letra maiúscula; apenas o título em
negrito, o subtítulo não fica em negrito)

PAU DOS FERROS
2025