



Fundamentos de Sistemas Operacionais

Trabalho 06

Prof. Tiago Alves

Gerência de Memória

Introdução

A disciplina de Fundamentos de Sistemas Operacionais trata de diversos tópicos desses sistemas que provêm uma forma intuitiva de se utilizar as funcionalidades de computadores digitais sem que seja necessário ao usuário ou programador ter profundo conhecimento das interações entre os diferentes *hardwares* que compõem um computador.

Para construir ou adicionar funcionalidades a esses sistemas computacionais, é necessário conhecimento de linguagens de programação e ferramentas de desenvolvimento. Em nosso curso, o domínio da linguagem C é um pré-requisito para o devido acompanhamento das atividades da disciplina.

Objetivos

- 1) Exercitar conceitos da linguagem de programação C, especialmente aqueles referentes à programação de sistemas operacionais.
- 2) Exercitar aspectos de programação de sistemas operacionais referentes a algoritmos de substituição de páginas.

Referências Teóricas

Mitchell, Mark, Jeffrey Oldham, and Alex Samuel. Advanced linux programming. New Riders, 2001.

Material Necessário

- Computador com sistema operacional programável
- Ferramentas de desenvolvimento GNU/Linux ou similares: compilador GCC, depurador, editor de texto.

Roteiro

- 1) Revisão de técnicas e ferramentas de desenvolvimento usando linguagem C.

Colete o material acompanhante do roteiro do trabalho a partir do Moodle da disciplina e estude os princípios e técnicas de desenvolvimento de aplicações usando linguagem C e sistema operacional Linux.



- 2) Realizar as implementações solicitadas no questionário do trabalho.

Nas referências bibliográficas que acompanham o trabalho há uma breve apresentação de ferramentas de depuração para o ambiente Linux.

ATENÇÃO: Suas implementações deverão ser construídas a partir de um Makefile.

Implementações e Questões para Estudo

- 1) Escreva um programa em C que simule o funcionamento dos principais algoritmos de substituição de páginas usados em sistemas operacionais.

- Os algoritmos cuja implementação é esperada são:
 - FIFO (First In, First Out);
 - OTM (Algoritmo Ótimo);
 - LRU (Least Recently Used ou Menos Recentemente Utilizado).
- Entrada:
 - O seu programa deverá ler da entrada padrão uma sequência de número inteiros onde o primeiro número representa a **quantidade de quadros de memória disponíveis** na RAM e os demais representam a sequência de referências às páginas, sempre um número por linha.
 - O número máximo de entradas em relação a referência de páginas não será superior a 30.
 - A entrada é composta por uma série números inteiros, um por linha, indicando, primeiro a quantidade de quadros disponíveis na memória RAM e, em seguida, a sequência de referências à memória. Entrada igual a -1 indica final de entrada. Exemplo em que o sistema dispõe de apenas 4 quadros de memória:

```
./trabalho_06
```

```
4
1
2
3
4
1
2
5
1
2
3
4
5
-1
```

- Saída:
 - Seu programa deverá imprimir na saída o número de “faltas de páginas” obtido com a utilização de cada um dos algoritmos.
 - A saída é composta por linhas contendo a sigla de cada um dos três algoritmos e a quantidade de “faltas de página” obtidas com a utilização de cada um deles. Exemplo:
FIFO 10
OTM 6



LRU 8

- Para a correção dessa questão será usado um juiz eletrônico. Atentem-se quanto ao formato de entrada (stdin) e quanto a eventuais erros de formatação de saída.

Instruções e Recomendações

A submissão das respostas aos problemas dos trabalhos deverá ser feita através do Moodle da disciplina.

Cada Problema do Trabalho 06 deverá ser entregue em um pacote ZIP. A dupla de alunos deverá nomear o pacote ZIP da seguinte forma: nome_sobrenome_matricula_nome_sobrenome_matricula_**trab06.zip**.

Entre os artefatos esperados, listam-se:

- códigos-fonte C das soluções dos problemas;
- documentação mínima da aplicação:
 - o qual sistema operacional foi usado na construção do sistema;
 - o qual ambiente de desenvolvimento foi usado;
 - o quais são as telas (instruções de uso)
 - o quais são as limitações conhecidas

Não devem ser submetidos executáveis.

Códigos-fonte C com erros de compilação serão desconsiderados (anulados).

Os trabalhos poderão ser realizados em duplas; a identificação de cópia ou plágio irá provocar anulação de todos os artefatos em recorrência.