

Trabalho Prático 1 – Simulador de Memória Cache**Valor: 10 pontos****Data de entrega: 14/05/2018****Introdução**

O objetivo deste trabalho prático é familiarizar o aluno com os principais conceitos estudados sobre Hierarquia de Memória. Foram estudados três tipos de projeto de organização da cache (Mapeamento direto, associativo por conjunto e totalmente associativo). Este trabalho destina-se a utilizar de forma prática esses projetos de cache. Dessa forma, será necessário a implementação desses tipos de cache e a análise da execução para grupos de entradas previamente fornecidos.

1. O simulador

Faça um simulador de memória cache que tenha os seguintes tipos de projeto de cache:

- a) Mapeamento direto;
- b) Associativo por conjunto (2 e 4 vias);
- c) Totalmente associativo.

As políticas de substituição (LRU, FIFO e LFU) da cache devem ser implementadas para os tipos de projeto que necessitarem de uma política de substituição;

A Figura1 exemplifica de forma geral o comportamento do simulador que deve ser implementado.

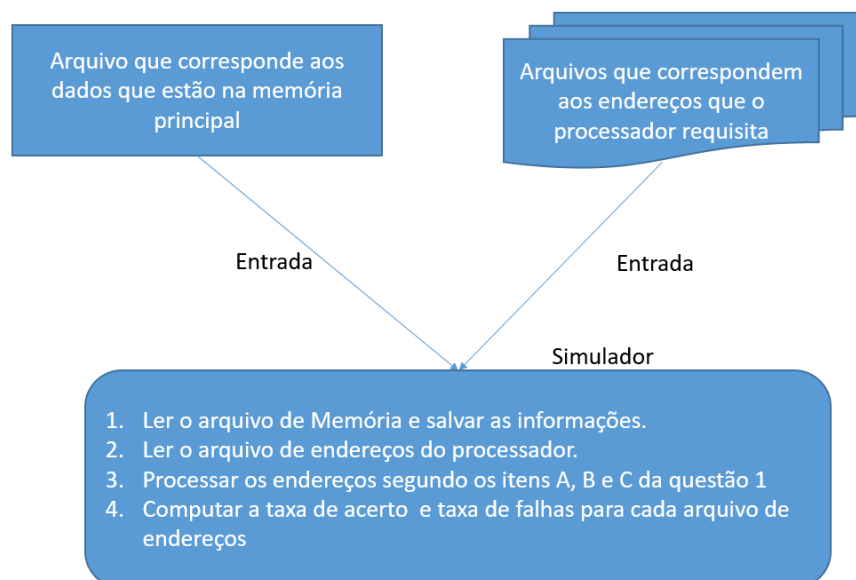


Figura 1: Requisitos do Simulador

2. O experimento

O experimento deverá conter uma análise das taxas de falha e acerto para grupo de arquivo. Deverão ser criados gráficos para cada grupo de arquivos, para visualização do comportamento dos diferentes tipos de projetos de cache de cada grupo de arquivos.

A memória principal tem o mesmo tamanho para cada grupo, 64 posições, porém, ela é organizada de forma distinta para atender cada projeto de cache.

Os programas em cada grupo (exemplo: Prog1.1 = Prog2.1 = Prog3.1 = Prog4.1 = Prog5.1) são iguais sendo as instruções podendo ser organizadas de forma diferente para atender cada grupo;

No seu relatório deverá ser feita uma análise entre os programas de cada grupo em cada tipo de projeto de cache.

- Como por exemplo para o mapeamento direto prog1: terá que ser comparado (Prog1.1, Prog2.1, Prog3.1, Prog4.1e Prog5.1)

Grupo 1: Cache de 2 linhas com endereços de 6bits (5 para TAG e 1 para índice)
Arquivos:

- M1
- Prog1.1
- Prog1.2
- Prog1.3
- Prog1.4
- Prog1.5

Grupo2: Cache de 4 linhas com endereços de 6bits (4 para TAG e 2 para índice)
Arquivos:

- M2
- Prog2.1
- Prog2.2
- Prog2.3
- Prog2.4
- Prog2.5

Grupo3: Cache de 8 linhas com endereços de 6bits (3 para TAG e 3 para índice)
Arquivos:

- M3
- Prog3.1
- Prog3.2
- Prog3.3
- Prog3.4
- Prog3.5

Grupo4: Cache de 16 linhas com endereços de 6bits (2 para TAG e 4 para índice)
Arquivos:

- M4
- Prog4.1
- Prog4.2
- Prog4.3
- Prog4.4
- Prog4.5

Grupo5: Cache de 32 linhas com endereços de 6bits (1 para TAG e 5 para índice)
Arquivos:

- M5
- Prog5.1
- Prog5.2
- Prog5.3
- Prog5.4
- Prog5.5

Entre outras coisas, a documentação deve conter:

1. Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do simulador.
2. Implementação: descrição sobre a implementação do simulador. Devem ser detalhadas as estruturas de dados utilizadas (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, bem como decisões tomadas relativas aos casos e detalhes que porventura estejam omissos no enunciado.
4. Experimento: Nessa parte do trabalho deverá ser feita uma análise sobre a taxa de acerto e falhas na cache. Por meio de gráficos deverão ser comparadas as taxas de acerto e falha de cada programa em cada projeto de cache e grupo de instância.
5. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
6. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso.

O que deve ser entregue:

Envie um arquivo ZIP com o nome no formato 'tp1-primeironome1-primeironome2.zip', contendo os seguintes arquivos:

- Arquivo README.txt com os nomes completos dos alunos do trio.

- O código fonte do programa;
- Relatório da observação das entradas.
- O trabalho deverá ser enviado para o e-mail: fischerbatera@hotmail.com, lembre-se que você após enviar deverá receber uma confirmação.

Comentários Gerais:

- Comece a fazer este trabalho logo, enquanto o problema está fresco na memória e o prazo para terminá-lo está tão longe quanto jamais poderá estar.
- Clareza, endentação e comentários no programa também serão avaliados.
- O trabalho deverá ser feito em dupla.
- Trabalhos copiados serão penalizados conforme anunciado.
- Penalização por atraso: $(2^d - 1)$ pontos, onde d é o número de dias de atraso.