

ARQUITETURA DE COMPUTADORES

TRABALHO DE AV2 – 2021

Competências avaliadas

- Compreensão do uso de hierarquia de memória para melhoria de desempenho de computadores.
- Compreensão de mapeamento direto de dados em memória Cache.
- Compreensão de mapeamento associativo de dados em memória Cache.
- Cálculo de parâmetros importantes para realização de mapeamento de memória.

Atividade

Escreva um programa, na linguagem que preferir, que ajude na compreensão dos métodos de mapeamento direto em cache e mapeamento associativo. Para isso, seu programa deve atender aos seguintes critérios mínimos:

Parte 1 – Configuração das memórias

Entradas

Usuário decide se irá realizar um mapeamento de memória direto ou associativo.

Usuário informa o tamanho da memória principal em Bytes.

Usuário informa o tamanho da memória Cache em Bytes.

Usuário informa o tamanho de cada Bloco/Linha em Bytes.

Saídas

O programa informa o mapeamento escolhido.

O programa retorna a quantidade de bits utilizados na TAG, LINHA e PALAVRA. Caso o mapeamento seja associativo, informar 0 bits para LINHA.

O programa retorna o valor hexadecimal de cada campo; TAG, LINHA e PALAVRA.

Na memória cache, inicie as tags com um valor -1, para indicar que não existe uma tag válida.

Parte 2 – Cálculos de Cache Hit e Cache Miss

Entradas

O usuário entra com uma sequência de endereços em hexadecimal, separados por um espaço, simulando uma sequência de requisições do processador a endereços da memória principal.

Saídas

O programa fornece a quantidade de Cache Hits seguida da sequência de endereços que geraram Cache Hit.

O programa fornece a quantidade de Caches Misses seguida da sequência de endereços que geraram Cache Miss.

Recomendações

A memória Cache pode ser simulada por uma variável composta ou pode ser criada como um objeto (no caso de uso de Orientação a Objetos).

A memória Principal pode ser simulada por uma variável composta ou pode ser criada como um objeto (no caso de uso de Orientação a Objetos).

Gere dados aleatórios de 0 a 255 para preencher cada posição da memória principal e da memória cache quando um programa é iniciado.

A *tag* da memória cache é um campo à parte que precisa ser armazenado, além do endereço e dos dados. Já na memória principal, a *tag* é obtida do endereço requisitado, como já estudamos.

No mapeamento associativo, utilize política LRU para decidir sobre os dados a serem substituídos.

Não se esqueça da máxima: programe um pouco e teste um pouco.