

# Arquitetura de Computadores

## Especificação do Segundo Trabalho Prático

### Simulador de Caches

O trabalho consiste na construção de um simulador de caches e por meio dele, avaliar distintas configurações de memórias caches. Mais especificamente, os objetivos são:

- construir um modelo de cache considerando três parâmetros de entrada:
  - quantidade de blocos de cache;
  - quantidade de words por bloco de cache;
  - associatividade da cache;
- executar um conjunto de operações de leitura e escrita sobre essa cache, utilizando o arquivo de entrada *trace.in*;
- devolver como resposta (na tela ou em arquivo de saída):
  - a *taxa de acerto* ou *hit rate*;
  - a quantidade de vezes que precisamos ir à memória principal para leitura ou para escrita;

## 1 Descrição do arquivo de entrada

O arquivo de entrada deve ter o seguinte formato:

```
n_blocks      // quantidade total de blocos da cache
               // (inteiro e potência de 2)
block_size    // quantidade de words por bloco de cache
               // (inteiro e potência de 2)
assoc         // associatividade da cache (ou, equivalentemente,
               // a quantidade de blocos por conjunto)
               // (inteiro e potência de 2)
n             // quantidade de operações que serão simuladas
               // (inteiro)
```

Em seguida, o arquivo contém *n* linhas, cada uma descrevendo uma operação que deve ser executada sobre o modelo de cache. As operações podem ser de leitura (0) e de escrita (1). Para cada operação deve ser passado

o endereço de memória principal (para leitura ou escrita). Caso a operação seja de escrita, deve haver um terceiro valor na linha, que é o dado que deve ser escrito no endereço previamente passado. Exemplo:

```
0 4096    // leitura no endereço 4096
0 128     // leitura no endereço 128
0 256     // leitura no endereço 256
0 1000    // leitura no endereço 1000
1 4096 0   // escrita do valor 0 no endereço 4096
1 256 2    // escrita do valor 2 no endereço 256
1 256 4    // escrita do valor 4 no endereço 256
...
```

Cada operação descrita nesse arquivo refere-se a um pedido feito pelo processador. Como temos um modelo de cache entre o processador e a memória principal, tentaremos encontrar na cache o bloco correspondente ao endereço passado pelo processador. Se o bloco que contém o dado solicitado está na cache, aumentamos uma unidade o contador de *hits* ou *acertos*. Se o dado não está na cache, contamos um *miss* ou *falha*.

**Por simplicidade, utilizem política de escrita *write-through* e política de substituição de blocos *fifo*.**

## 2 Entrega e Avaliação

- O trabalho pode ser feito individualmente ou em duplas;
- Pode-se utilizar linguagem de programação C, C++ ou Java;
- A entrega será no dia 12/09, até às 23h, pelo moodle;
- Deve-se entregar um arquivo único (.tar.gz) contendo todos os fontes e arquivos de entrada. Deve-se incluir também um arquivo de texto chamado README contendo as instruções para compilação e execução do seu programa;

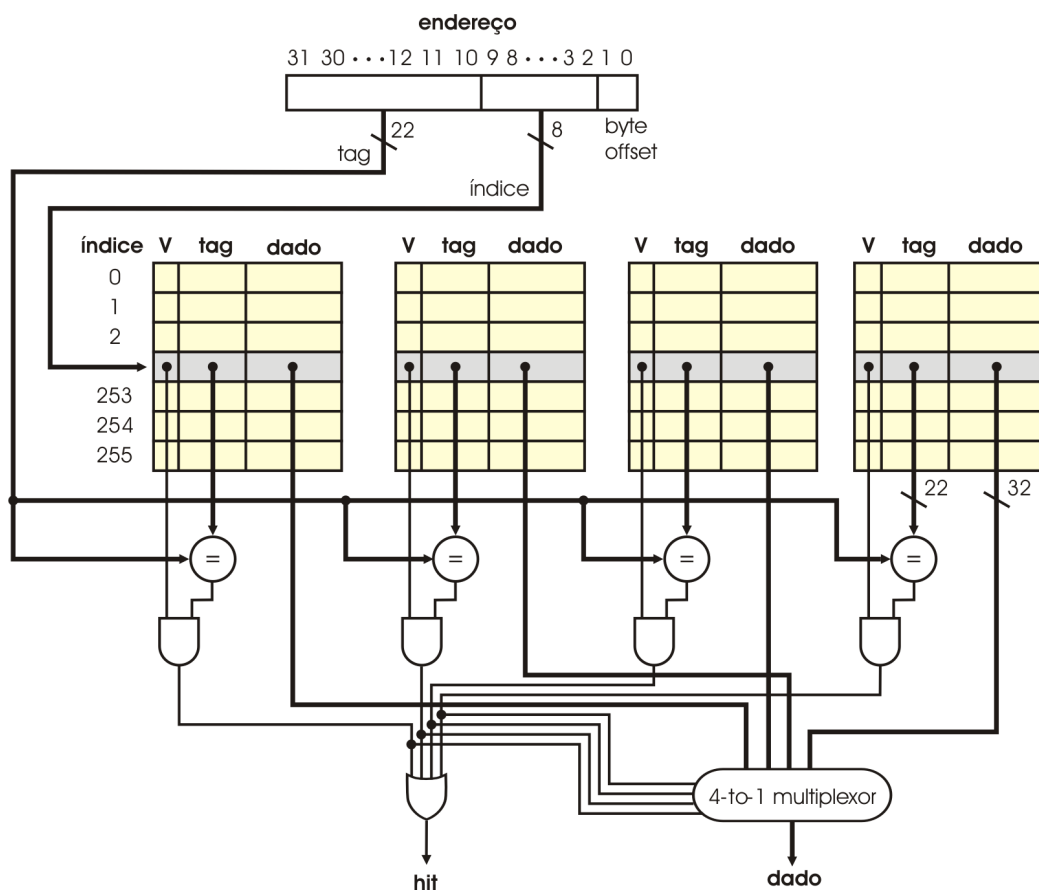


Figura 1: Exemplo de uma memória cache com 1024 blocos, associatividade 4, 1 word por bloco