

Formas de armazenamento de dados em memória

A questão do armazenamento é um assunto pertinente na história humana, que se estende desde armazenamento de comida, água e materiais essenciais à sobrevivência até atualmente em que o armazenamento seguro e eficiente de dados em memória de forma digital. Neste capítulo nos ateremos aos seguintes modos de armazenamento de dados: Definido por Software, Nuvem, Rede, de objetos, de arquivos e em blocos.

Métodos de acesso aos dados na memória

As diferentes memórias dentro de um computador podem ser classificadas dependendo de suas características. A Tabela 1 mostra as principais características dos sistemas de memória de um computador (STALLINGS, 2010). Neste capítulo iremos abordar apenas os quatro métodos de acesso aos dados na memória.

Localização Interna (por exemplo, registradores do processador, memória principal, cache) Externa (por exemplo, discos ópticos, discos magnéticos, fitas)	Desempenho Tempo de acesso Tempo de ciclo Taxa de transferência
Capacidade Número de palavras Número de bytes	Tipo físico Semicondutor Magnético Óptico Magneto-óptico
Unidade de transferência Palavra Bloco	Características físicas Volátil/não volátil Apagável/não apagável
Método de acesso Sequencial Direto Aleatório Associativo	Organização Módulos de memória

Tabela 1 - Principais características dos sistemas de memórias (STALLINGS 2010).

Os métodos de acesso aos dados na memória são divididos em:

Acesso sequencial: unidades de registro compõem a organização da memória, separados por endereços. O acesso é feito em uma sequência linear específica, onde a informação de endereçamento é usada para separar registros e auxiliar na recuperação dos dados. Neste método de acesso, é utilizado um mecanismo de leitura-escrita compartilhado, que deve ser movido ao local desejado, ignorando registros no caminho (registros intermediários). Por este motivo, um dado que deve ser acessado mais próximo de onde o usuário quiser ler ou escrever, demora menos tempo. Assim, o tempo de acesso a um registro é extremamente variável. Um exemplo de dispositivo de memória que utiliza o acesso sequencial são as fitas magnéticas. A Figura 1 exemplifica o funcionamento de uma fita com acesso sequencial (STALLINGS,2010).



Acesso direto: no método de acesso direto, assim como no método sequencial, as memórias são organizadas em registros, mas separadas por endereços específicos, baseados no local físico. Também envolve um mecanismo de leitura-escrita compartilhado. O acesso é realizado pelo acesso direto, para alcançar uma vizinhança geral, mais uma busca sequencial, contagem ou espera, até alcançar o local final.(STALLINGS, 2010). Assim como no acesso sequencial, o tempo de acesso aos dados torna-se variável. Um exemplo de dispositivo que utiliza o acesso direto são as unidades de disco. A figura 2 mostra um exemplo de disco com cinco zonas (setores), onde cada zona possui várias trilhas (TANENBAUM, 2013).

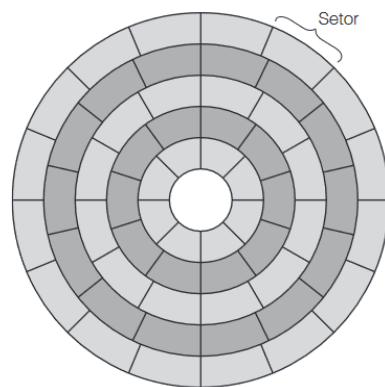


Figura 2 -Disco com 5 zonas.

Acesso aleatório: diferente dos métodos anteriores, no acesso aleatório, cada posição na memória possui um endereço exclusivo, conectado fisicamente. Por este motivo, o tempo para acessar determinado local é independente da sequência de acessos anteriores e é constante. Assim, qualquer local pode ser selecionado aleatoriamente, e endereçado e acessado diretamente. (STALLINGS, 2010). A memória principal (RAM), onde o próprio nome já leva a definição de aleatório (Random Access Memory), é um exemplo de acesso aleatório. A figura 3 ilustra isso.

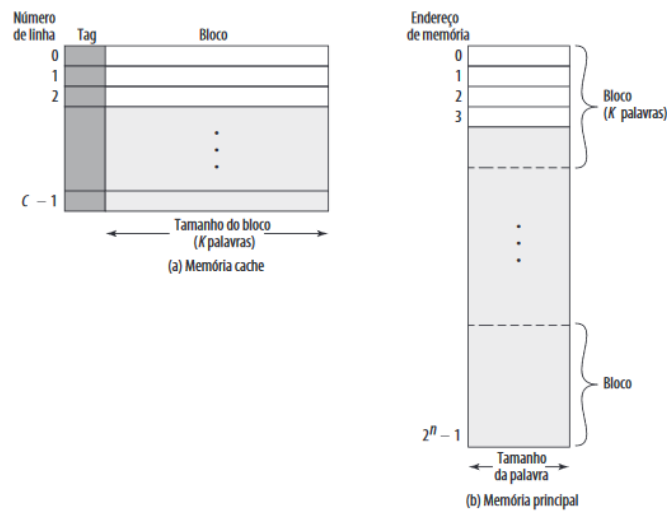


Figura 3 - Estrutura de cache/memória principal

Acesso associativo: esse é o tipo de memória de acesso aleatório que permite fazer uma comparação de um certo número de bit desejados dentro de uma palavra para uma combinação especificada, e faz isso para todas as palavras simultaneamente. Assim, uma palavra é recuperada com base em uma parte de seu conteúdo, em vez do seu endereço (STALLINGS, 2010). Assim como a memória de acesso aleatório comum, o tempo de recuperação é constante, independente do local acessado. Um exemplo de memória que utiliza o acesso associativo são as memórias cache. A figura 4, por fim, ilustra um mapeamento associativo (STALLINGS, 2010).

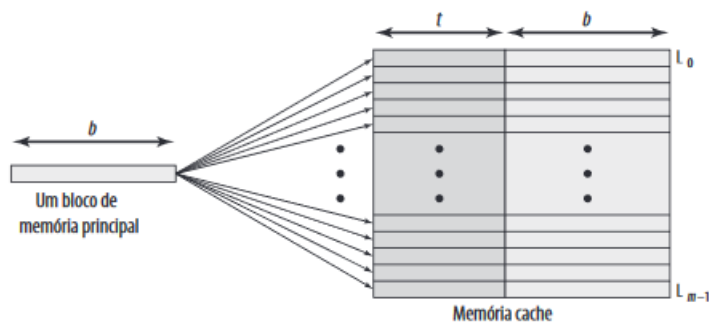


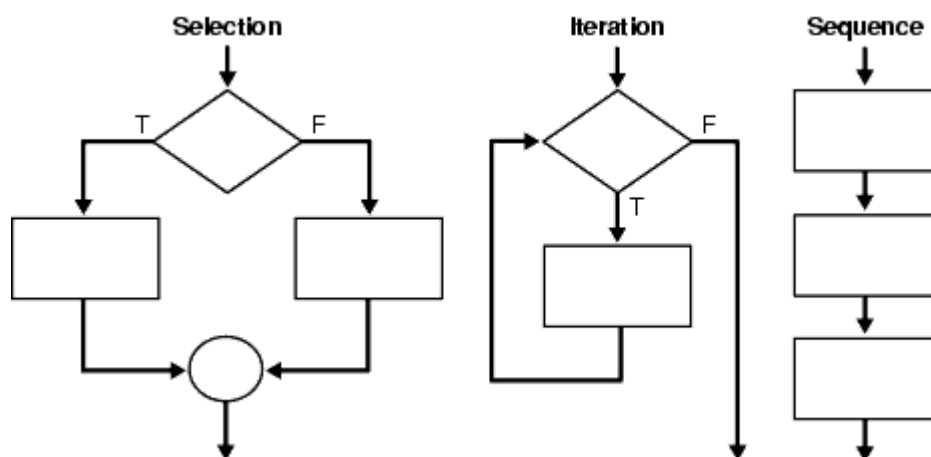
Figura 4 - mapeamento associativo memória cache.

Estrutura de controle da linguagem de máquina

Uma Estrutura de Controle é um bloco de programação que analisa variáveis e escolhe uma direção para seguir baseado nos parâmetros pré-definidos. O termo controle de fluxo define bem a sua função: sendo nada mais do que o processo básico de tomada de decisão das máquinas, o controle de fluxo determina como um software ou uma aplicação responderá diante de certas condições e parâmetros.

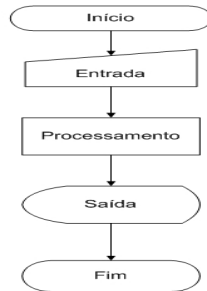
Em nível de linguagem de máquina, as instruções de estruturas de controle geralmente funcionam ao alterar o contador de programa. Um contador de programa (PC) é um registro da CPU no processador do computador que possui o endereço da próxima instrução a ser executada da memória. O contador do programa funciona em combinação com outros registradores para identificar a instrução atual. Ele pode ser modificado ou acessado com a ajuda das instruções de acesso ou salto. O PC pode ser acessado, modificado por instruções de salto e desvio. Portanto, o endereço de destino pode ser carregado no contador do programa por meio de instruções de desvio. O contador do programa também pode ser carregado com o endereço usando as instruções de processamento de dados. Para algumas CPUs, as únicas instruções de estruturas de controle disponíveis são os diversos tipos de jump condicional.

Uma vez compreendida a função da estrutura de controle na programação, pode-se chegar à conclusão de que, independentemente do projeto, a maneira como planejamos e executamos sempre segue uma estrutura de controle. As estruturas de controle têm basicamente três tipos: Estrutura sequencial, Estrutura de decisão e Estrutura de repetição. A figura abaixo ilustra por meio de um fluxograma os três tipos de estrutura de controle.



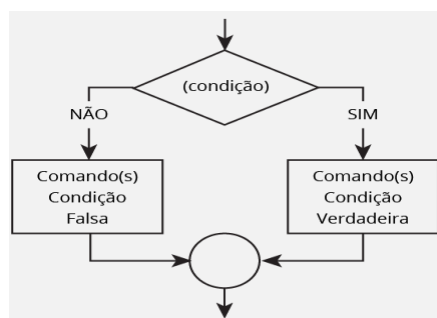
Estrutura sequencial: Na estrutura sequencial os comandos de um algoritmo são executados numa sequência pré-estabelecida. Cada comando é executado somente após o término do comando anterior. Em termos de fluxogramas, a estrutura sequencial é caracterizada por um único fluxo de execução (um único caminho orientado) no diagrama.

Implementação da estrutura Sequencial

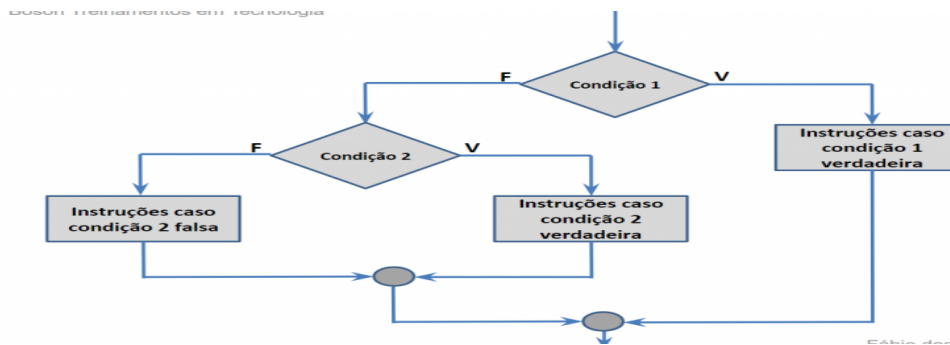


Estrutura de Decisão: Neste tipo de estrutura o fluxo de instruções a ser seguido é escolhido em função do resultado da avaliação de uma ou mais condições. Uma condição é uma expressão lógica. A classificação das estruturas de decisão é feita de acordo com o número de condições que devem ser testadas para que se decida qual o caminho a ser seguido. Segundo esta classificação, têm-se dois tipos de estruturas de decisão: Se e escolha.

Estruturas de Decisão do Tipo Se: Nesta estrutura uma única condição (expressão lógica) é avaliada. Se o resultado dessa avaliação for verdadeiro (.V.), então um determinado conjunto de instruções (comando composto) é executado. Caso contrário, ou seja, quando o resultado da avaliação for falso (.F.), um comando diferente é executado.

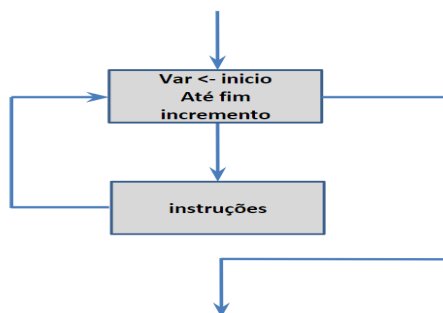


Estruturas de Decisão do Tipo Escolha: Este tipo de estrutura é uma generalização da estrutura Se, onde somente uma condição era avaliada e dois caminhos podiam ser seguidos. Na estrutura de decisão do tipo Escolha pode haver uma ou mais condições a serem testadas e um comando composto diferente associado a cada uma destas.



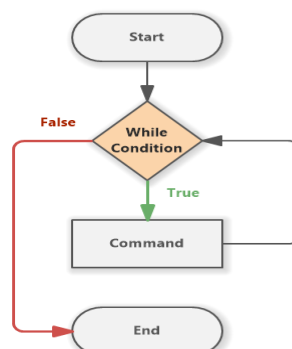
Estruturas de Repetição: é uma estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição ou com um contador. São utilizadas, por exemplo, para repetir ações semelhantes que são executadas para todos os elementos de uma lista de dados, ou simplesmente para repetir um mesmo processamento até que a condição seja satisfeita. As estruturas de repetição são muitas vezes chamadas de Laços ou, também, de Loops. Existem dois tipos de laço: laços contados e laços condicionais.

Laços Contados: Os laços contados são úteis quando se conhece previamente o número de vezes que se deseja executar um determinado conjunto de comandos. Então, este tipo de laço nada mais é que uma estrutura dotada de mecanismos para contar o número de vezes que o corpo do laço (ou seja, o comando composto em seu interior) é executado.

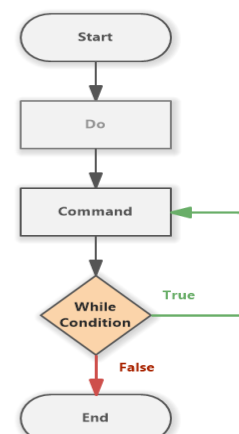


Laços Condicionais: São aqueles cujo conjunto de comandos em seu interior é executado até que uma determinada condição seja satisfeita. Ao contrário do que acontece nos laços contados, nos laços condicionais não se sabe de antemão quantas vezes o corpo do laço será executado. As construções que implementam laços condicionais mais comuns nas linguagens de programação modernas são: Enquanto e Repita.

WHILE



DO-WHILE



Referências bibliográficas

STALLINGS, William. Arquitetura e Organização de Computadores. 8ª ed. São Paulo: Pearson, 2010.

TANENBAUM, Andrew S. Organização estruturada de computadores. 6ª ed. São Paulo: Pearson Prentice Hall, 2013.