

Arquitetura de Computadores

Prof. Frederico Sauer

<http://www.fredsauer.com.br>
fsauer@gmail.com

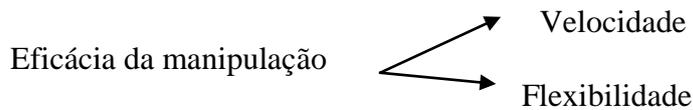
Livro Texto Adotado:

Introdução a Org. de Computadores – 4ª edição - Mário A. Monteiro
LTC

Introdução: Dados → Processamento → Informação

Computar = calcular. Computadores são equipamentos capazes de realizar cálculos.

Evolução – Aumento do volume de dados



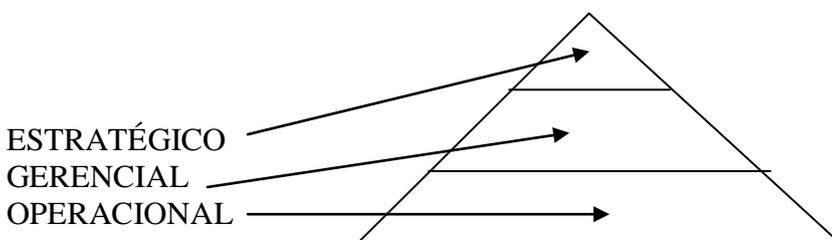
Sistemas

Sistema : “conjunto de partes coordenadas que concorrem para a realização de um objetivo”

Sistemas de PD: coleta, armazenamento, processamento e recuperação de dados necessários a um sistema de informações. É composto por um sistema de computação (computador e o software básico) e os sistemas de aplicações.

Sistema de informações: métodos, processos e equipamentos necessários a obter processos e permitir a utilização de informações dentro da empresa.

Hierarquia dos Sistemas de Informações



Cada nível, por possuir diferentes competências, demandam diferentes tipos de modelos para apresentação e uso da informação. O que para um nível é apenas um dado, para outro nível pode representar uma informação. Ex.: valor total de uma folha de pagamento.

Arquiteturas Computacionais

SISD – Single Instruction stream, Single Data stream – Processadores executam uma única instrução completa de cada vez – máquina de Von Neumann – não é mais utilizado

MISD – Multiple Instruction stream, Single Data stream – Um único conjunto de dados que pode ser utilizado simultaneamente com execução simultânea – Ex.: processadores vetoriais

SIMD – Single Instruction stream, Multiple Data stream – Muito usada nos dias atuais. O *Pipelining* dos PC é um exemplo.

MIMD - Multiple Instruction stream, Multiple Data stream – Máquinas multiprocessadas, *clusters* de computadores.

Sistema de Computação

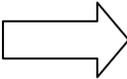
Programa – etapas manuais ou automáticas elaboradas passo – a – passo.

Passo – instrução (ordem de comando) dada ao Hardware para a realização de uma determinada ação

Logo, um programa é um conjunto de instruções

Ex: sistema p/somar 100 números - máq. calcular + pessoa + máq. Escrever

Tarefas – algoritmo (passos finitos e ordenados a executar)

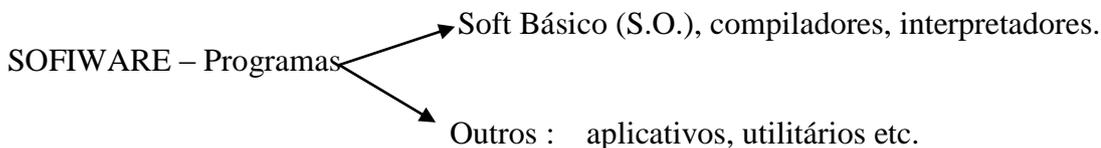
1. Escrever e guardar $N=0$ e $SOMA = 0$
2. Ler n° entrada
3. Somar n° com $SOMA$ e guardar em $SOMA$
4. Somar 1 a N e guardar em N
5. SE N for < 100 , ir para o passo 2  Teste de parada – garante que é finito
6. Imprimir $SOMA$
7. Parar

Programa de Computador – transformação de um algoritmo em uma linguagem inteligível pelo computador a executá-lo em forma de linguagem de máquina. Dados, instruções e resultados são zeros e uns.

Linguagem – regras fixas e rígidas de sintaxe

Programas escritos em linguagem (Pascal, Cobol, C, C++) precisam ser convertidos para linguagem de máquina antes de serem executados.

HARDWARE – circuitos elétricos e dispositivos eletromecânicos do computador



HISTÓRICO

Fósseis dos homens das cavernas – possíveis formas de computar objetos (posses)

V a.c. - Abaco

1642 – Pascaline – soma e subtrações (soma ao complemento a 10)

1671 – Leibnitz – multiplicações e divisões

1801 – Tábuas de Jacquard na tecelagem

1823 – Charles Babagge – Analytical Engine – primeira máquina capaz de executar um programa em cartões perfurados. Influência de Ada Lovelace (STORE, JUMP, etc)

1890 – Hollerith – cartões perfurados – primeira máquina eletromecânica

$\begin{array}{l} 7-2= 7+8 \text{ (resto) } \underline{5} \\ 7+(3+\underline{5}) \end{array}$

censo 1880 levou 10 anos e o de 1890 levou 2 anos e ½ - TMC depois IBM
1935 – Zuze – máquina de calcular eletrônica com relés mecânicos usando bits ao invés de algarismos decimais, usados nas máquinas de Babbage.
1937 – Máquina de Turing – definição de uma função de computação e a completa descrição de uma teoria de Computação.
1945 – 1º Comp. Eletronico – Válvulas substituem os relés - ENIAC (Eletronic Numerical Integrator and Calculator) – 30 toneladas e 17 mil válvulas (levava 1 dia p/computar cálculos manuais de 300 dias)
1945 – Jonh Von Newmann – Arquitetura Computacional com programa armazenado. Memória principal, registradores, UCP, etc

Década 50 – Transistor (Bell Laboratories). Executa as mesmas funções que uma válvula, mas com menor consumo e dissipação – IBM 1401 – grande sucesso comercial

1964 – IBM /360 (família mais famosa da IBM) – circuitos integrados – acomodação de transistores, capacitores e resistores num único componente. Um único componente de silício pode representar funções de múltiplos transistores (chip)

Década 70 – Integração de CI em pastilhas (chips) através de VLSI (Very Large Scale Integration) – milhões de componentes num único chip. Possibilitou o surgimento dos PC.

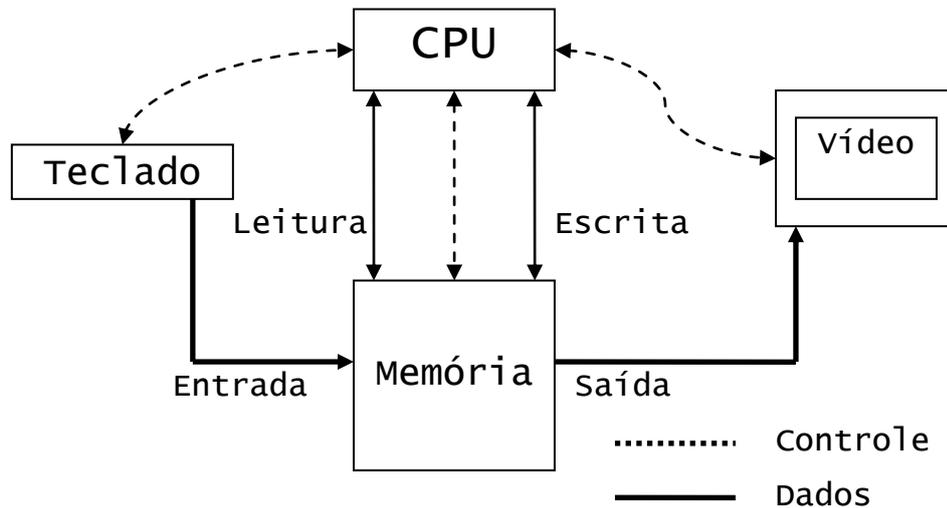
Evolução das Linguagens

1º geração – Linguagem de máquina (zeros e uns).
2º geração – Assembly – siglas inteligíveis como ADD, STORE, JUMP, etc.
3º geração – (alto nível) – Fortran, Cobol, Basic, Pascal
4º geração – SQL, Dbase – conversacional

Windows Kits – Delphi, Visual Basic – ambiente de desenvolvimento
Ambientes atuais – Java, DOT NET, C#, php, asp

Componentes de um Sistema de Computação

Primeiro passo no desenvolvimento de um sistema – construção de um algoritmo em linguagem pseudo-natural. Após isso o algoritmo é codificado em uma linguagem de programação, originando um **programa**. Os comandos deste programa precisam ser introduzidos no hardware, para que possam ser interpretados – **ENTRADA** – teclado, mouse, scanner, unidades de disco, etc podem ser usados para essa função. Junto com os dispositivos (periféricos) para saída do sistema compõe as unidades de E/S (entrada e saída). Para entender e realizar uma instrução de máquina usa-se a CPU (*Central Processing Unit*). Para que isso ocorra, é necessário que os dados necessários a este processamento estejam disponíveis no componente de ENTRADA memória. Uma vez processados os dados, as informações obtidas são disponibilizadas para o usuário através de um dispositivo de SAÍDA (display, impressora, etc)



Representação das Informações

Internamente, os computadores manipulam os dados, instruções e informações em forma eletrônica, ou seja, através de um valor de voltagem, corrente, magnética ou ótica. O uso dos bits deve-se a isso, pois é mais fácil implementar-se num equipamento que tem a capacidade de armazenar e operar usando funções LIGA/DESLIGA, ABERTO/FECHADO, SIM/NÃO, etc. O fundamento matemático é o da lógica booleana.

Língua humana – unidade atômica de informação - caractere (26 alfabéticos, 10 numéricos, sinais de pontuação e de operação matemática e lógica)

Língua Computacional – bit (binary digit) – 0 e 1

De acordo com o tamanho escolhido para grupos de bits (palavra) para codificar os caracteres, tem-se o número máximo de caracteres codificáveis (ex.: 5 bits – 32 caracteres)

IBM – uso dos 8 bits para armazenamento e transferência – *byte*. Até hoje essa referência é bastante usada (ex.: impressoras – recebe caractere por caractere -.byte a byte; memória – HD de 500 Gigabytes)

Kilo, Mega e Giga – em grandezas métricas, trata-se de multiplicar por 1000 (ex.: 1 kilo de café = 1000 gramas de café). No mundo binário, é diferente. Um Kilo “de bytes” não são 1000 bytes, e sim $2^{10} = 1024$ bytes.

Em taxas de transmissão, no entanto, Kilo, Mega, Giga significam, respectivamente 1000bps, 1000Kbps e 1000Mbps.

$$2^{10} \times 2^{10} = 2^{20} = 1.048.576 \text{ bytes, ou } 1024 \text{ Kbyte, ou } 1 \text{ Megabyte}$$

$$2^{10} \times 2^{10} \times 2^{10} = 2^{30} = 1.073.741.824 \text{ bytes, ou } 1024 \text{ Megabytes, ou } 1 \text{ Gigabyte}$$

$$2^{10} \times 2^{10} \times 2^{10} \times 2^{10} = 2^{40} = 1024 \text{ Gigabytes} = 1 \text{ Terabyte}$$

$$2^{10} \times 2^{10} \times 2^{10} \times 2^{10} \times 2^{10} = 2^{50} = 1024 \text{ Terabytes} = 1 \text{ Petabyte}$$

Assim, 128 Mbytes de memória = $128 \times 1024 \text{ bytes} = 131.072 \text{ bytes}$

Tipicamente, nos dias atuais, usa-se o byte para caracterizar a unidade de armazenamento, e a palavra (ex.: Pentium – 32 bits) para unidade de transferência e processamento.

Arquivos e Registros

Arquivo – conjunto de dados ou informações de um mesmo tipo. É estruturado em itens individuais, chamados de registros. Um programa também é um arquivo, porém de um único registro. A estrutura de armazenamento e recuperação de informações na memória secundária de um computador também é concebida na forma de arquivos e registros (diretórios e arquivos).

Classificação dos Sistemas de Computação

- ✓ Microcomputadores – uso pessoal. Abrangem desde os relógios com Linux até os PC
 - Desktops – micro de mesa
 - Netbook, Laptop ou Notebook – portátil
 - Palmtops – cabe na palma da mão
 - Gadgets – dispositivos processados pequenos usados para aplicações específicas – Game boy, celulares com funcionalidade computacional, etc.
- ✓ Workstation – microcomputador “turbinado” para execução de tarefas específicas, onde se precise de determinadas características dos microcomputadores em maior capacidade. Ex.: Estações de CAD (muita memória, monitores especiais, placas de vídeo apropriadas).
- ✓ Minicomputadores – diferentemente dos anteriores, prevê a utilização multiusuária e multiprogramação. Hoje quase não são mais usados, uma vez que os microcomputadores já podem exercer tal funcionalidade.
- ✓ Mainframes (computadores de grande porte) – criados para manipular quantidades gigantescas de dados, atendendo simultaneamente vários (centenas) usuários. Após um período de declínio, provocado pelos processos de *downsizing*, onde foram substituídos por microcomputadores ligados em rede, ainda possuem sobrevivência na área de controle aéreo, algumas aplicações comerciais e, mais recentemente, passa a ser rediscutido com a experiência de sucesso onde o Linux foi portado pela IBM para equipamentos de grande porte.
- ✓ Supercomputador – criado para atender ao propósito específico de realizar uma grande quantidade de cálculos matemáticos o mais rapidamente possível. Não se prestam a aplicações comerciais.

Medidas de Desempenho de Sistemas de Computação

O desempenho é geralmente medido através de sua velocidade de trabalho. Há vários parâmetros diferentes usados no mercado.

MIPS – milhões de instruções por segundo

MFLOPS – milhões de operações de ponto flutuante por segundo

SPEC (System Performance Evaluation Cooperative – HP, SUN, etc.) – 16 programas de teste e medida.

Tempo de resposta – medida de desempenho global, não considerando as diferenças de performance entre os vários componentes do sistema.

Vazão – Throughput – quantidade de ações que pode ser realizada pelo sistema em uma unidade de tempo. Depende do subsistema avaliado (ex.: taxa de transferência entre HD e barramento da placa-mãe, taxa de transferência entre a memória e o barramento da placa-mãe, etc.)

Conversão de Bases e Aritmética Computacional

Notação Posicional - usado para representação numérica

- os algarismos assumem valores diferentes de acordo com sua posição

ex: 1 – valor um

10 – o 1 representa o valor 10

Sistema decimal – dez símbolos, com $S=\{0,1,2,3,4,5,6,7,8,9\}$ – sistema de BASE 10

Representação

1303 (tradicionalmente dispensa-se o 10 da base)

$$1303 = 1 \times 10^3 + 3 \times 10^2 + 0 \times 10^1 + 3 \times 10^0$$

$$d_0 = 3$$

$$d_1 = 0$$

$$d_2 = 3$$

$$d_3 = 1$$

$$N = d_{n-1} \times b^{n-1} + d_{n-2} \times b^{n-2} + \dots + d_1 \times b^1 + d_0 \times b^0 \quad \text{fórmula A}$$

N – n° a expressar

D – cada algarismo do número

n – n° de algarismos inteiros

b – base de numeração

Representação de fracionários

$$N = d_{n-1} \times b^{n-1} + d_{n-2} \times b^{n-2} + \dots + d_0 \times b^0 + d_{-1} \times b^{-1} + d_{-2} \times b^{-2} + \dots + d_m \times b^{-m}$$

Parte inteira
Parte fracionária

Onde m é a quantidade de algarismos fracionários

Ex: $32,12_{10}$

$$= 3 \times 10^1 + 2 \times 10^0 + 1 \times 10^{-1} + 2 \times 10^{-2} = 30 + 2 + 0,1 + 0,02$$

Outras bases (binária, octal e hexadecimal):

Computador – digital (num dígito ou bit) com dois valores possíveis – 0 ou 1 (ligado ou desligado, falso ou verdadeiro)

Sistema de base 2 $\rightarrow S = \{0,1\}$ – números ficam muito grandes, demandando uso de outros sistemas como o octal (base 8), ou hexadecimal (base 16), pois na conversão para binário é mais simples e rápida

Com a fórmula da notação posicional, obtemos o valor decimal de um número qualquer expresso em qualquer base.

Exemplo: $(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 0 + 2 + 1 = (11)_{10}$

Vamos conferir?

0 – 0	11 – 3	110 – 6	1001 – 9	1100 – 12
1 – 1	100 – 4	111 – 7	1010 – 10	1101 – 13
10 – 2	101 – 5	1000 – 8	1011 – 11	1110 – 14

base hexadecimal $S = \{0,1,2, \dots, 9, A, B, C, D, E, F\}$

ex: $(1A7B)_{16} = 1 \times 16^3 + 10 \times 16^2 + 7 \times 16 + 11 \times 16^0$
 $4096 + 2560 + 112 + 11 = (6779)_{10}$

Em computadores, usam-se as bases 8 (octal) e 16 (hexadecimal) devido à facilidade e rapidez na conversão p/base 2 (binário) e vice-versa

Subtração binária – “pede emprestado”, só que, ao invés de 10, é 2 (o valor da base)

$$\begin{array}{r}
 2 \\
 011012 \\
 \text{Ex.: } 11001001 \\
 \underline{-10111011} \\
 00001110
 \end{array}
 \qquad
 \begin{array}{r}
 1202 \\
 100101 \\
 \underline{-11010} \\
 001011
 \end{array}$$

Aritmética Hexadecimal – “vai x” e “pede emprestado”

Hexadecimal	Decimal
A	10
B	11
C	12
D	13
E	14
F	15

$$\begin{array}{r}
 1 1111 \\
 \text{Ex.: } 94FFE A \\
 \underline{+9AE5FF} \\
 12FE5E9
 \end{array}
 \qquad
 \begin{array}{r}
 3B D \\
 4C7DE8 \\
 \underline{-E927A} \\
 3DEB6E
 \end{array}$$

contas da soma:

$$A(10) + F(15) = 25, =16 (\text{vai } 1) + 9$$

contas da subtração:

$$\begin{array}{r}
 16 \\
 \underline{+8} \\
 24 \\
 \underline{-10} \rightarrow A \\
 14 \rightarrow E
 \end{array}$$

$$\begin{array}{r}
 16 \\
 \underline{+7} \\
 23 \\
 \underline{-9} \\
 14 \rightarrow E
 \end{array}$$

$$\begin{array}{r}
 11 \rightarrow B \\
 \underline{+16} \\
 27 \\
 \underline{-14} \rightarrow E \\
 13 \rightarrow D
 \end{array}$$

Multiplicação binária – semelhantes e ainda mais simples do que na multiplicação decimal, pois:

$$0 \times 0 = 0, 0 \times 1 = 0, 1 \times 1 = 1 \text{ e } 1 \times 0 = 0.$$

$$\begin{array}{r}
 110 \rightarrow 6 \\
 \times 101 \rightarrow 5 \\
 \hline
 110 \\
 000 \\
 110 \\
 \hline
 11110 \rightarrow 30
 \end{array}$$

Divisão Binária – semelhante ao processo na base 10.

Ex.: $37 \div 5 = 7$, resto 2

Procedimento:

1. Verifica-se quantas vezes o dividendo (37) cabe no divisor (5), chegando-se ao número 7 (maior número que, multiplicado por 5, dá resultado igual ou inferior ao dividendo (37), obtendo-se o quociente 7 com resto 2.
2. Caso o resto ainda seja divisível pelo divisor, repete-se o procedimento agregando-se o resultado obtido ao quociente já apurado.

$101010 \div 110$

1. Verifica-se qual valor é suficientemente maior (em quantidade de algarismos) que o divisor (no caso, 110), de forma que o quociente comece com 1. No exemplo acima, o valor 1010 é maior uma vez que o divisor. Executando-se a divisão, temos:

$$\begin{array}{r} 101010 \mid 110 \\ - 110 \quad 1 \\ \hline 100 \end{array}$$

2. De forma similar à divisão decimal, ao resto obtido acrescenta-se o próximo algarismo do dividendo não utilizado na primeira operação e faz-se nova divisão pelo divisor.

$$\begin{array}{r} 101010 \mid 110 \\ - 110 \quad 11 \\ \hline 1001 \\ - 110 \\ \hline 011 \end{array}$$

3. Repete-se novamente o procedimento, até esgotar todos os bits do dividendo. Da mesma forma que na divisão decimal, pode-se ou não encontrar resto.

$$\begin{array}{r} 101010 \mid 110 \\ - 110 \quad 111 \\ \hline 1001 \\ - 110 \\ \hline 0110 \\ - 110 \\ \hline 000 \end{array}$$

4. Durante o aprendizado, é recomendável a conversão do binário para decimal e comparar os resultados.

$$\begin{array}{r} 101010 - 42 \\ 110 - 6 \end{array}$$

resultado obtido: 111, com resto zero. Em decimal, $7. 42 \div 6 = 7$ (ok !)

Agora, é só treinar bastante com os exercícios propostos no livro. Muitos alunos argumentam que não é importante para o profissional de computação o conhecimento da aritmética binária, octal e hexadecimal, o que é um grande equívoco. Pode ser exatamente a diferença entre um programador medíocre e um desenvolvedor conceituado no mercado, especialmente se estiver trabalhando com sistemas onde se precise de grande desempenho (todos eles !). Um dia, alguns de vocês se lembrarão disso, lamentando-se ou agradecendo por ter dedicado alguns minutos de empenho para compreender e aplicar estes conceitos.