

Sistemas Numéricos Computacionais

Agora que vamos estudar a área de informática, vimos um pouco de história do computador e também os conceitos de Hardware e Software.

Já ficou imaginando, "como os circuitos, processador, memória conseguem transformar algumas linhas de código em um programa ou aplicativo de celular".

Nós vemos números... todo o tempo

Sabemos que os números fazem parte do nosso dia a dia e a gente lida com eles no momento em que acorda até o sono chegar. O preço das coisas, horas trabalhadas, etc..

Com os números é possível representar muita coisa e também transformar essas representações numéricas em informações - ou até mesmo constituir uma forma de comunicação (pode ser um número de telefone, endereço, conta de banco, dentre outros exemplos).



Vamos pensar um pouco !!!

"Se nós, seres humanos, utilizamos os números para diversas abstrações do mundo real, por que com as máquinas seria diferente? "

Faz sentido então construir máquinas ou soluções de acordo com nossa compreensão da realidade, ou seja, que tenha alguma conexão com a nossa forma de organização do mundo real, e por isso utilizamos também números para dar instruções aos computadores.

Ok, chegamos a entender que comunicamos com as máquinas através de números.

Mas... as máquinas calculam números da mesma forma que a gente aprendeu na escola?

Agora vamos ater a um detalhe muito importante. Nós utilizamos 10 algarismos, os números 0,1,2,3,4,5,6,7,8,9 para criar, quantificar ou mesmo organizar qualquer coisa. É o chamado sistema decimal, são números que possuem base 10 e sua classificação está relacionada a posição que ocupa.

Agora parece que complicou? Vamos ver de outra forma.

Vejamos o número 125 em decimal:

10^3	10^2	10^1	10^0
1000	100	10	1
0	1	2	5

$$\begin{array}{r} 1 \times 5 = 5 \\ 10 \times 2 = 20 \\ 100 \times 1 = 100 \end{array}$$

125

Então resolvido, usamos sistema decimal para comunicarmos e programarmos computador. **NÃO**

Para a gente entender melhor esse ponto, vamos lembrar da breve história da computação que vimos nas primeiras aulas.

Nos primeiros computadores a programação era realizada com a conexão direta entre unidades físicas que enviavam os sinais elétricos, eram **válvulas** que você precisava trabalhar manualmente para poder enviar as instruções, você ligava e desligava para enviar as informações!

O MARK I e II e depois o ENIAC são exemplos de computadores de uso geral que precisavam ser ligados e desligados manualmente para ser programado. Eram milhares de válvulas que precisavam ser programadas manualmente.

Por conta dessas características, era difícil trabalhar e encontrar mão-de-obra para operar computadores como o ENIAC. Pensando nisso. Um dos consultores no projeto ENIAC foi o matemático Von Neumann, imaginou uma forma de facilitar o processo de programação com a proposta de um novo computador de programa armazenado em meados de 1940.

O EDVAC (Electronic Discrete Variable Automatic Computer, em tradução livre, calculadora eletrônica automática de variável discreta) e posteriormente o IAS (a sigla vem do nome do instituto de Princeton, Institute for Advanced Study - Instituto de estudos avançados) combinavam uma arquitetura que otimizava também o processamento de dados.

Matemáticos como Von Neumann e Alan Turing se apropriaram de conceitos de Leibniz (sistema binário de numeração), George Boole

(álgebra booleana) e aplicaram em seus estudos sobre arquitetura de computadores e criptografia, trabalhos que são a base da computação atual.



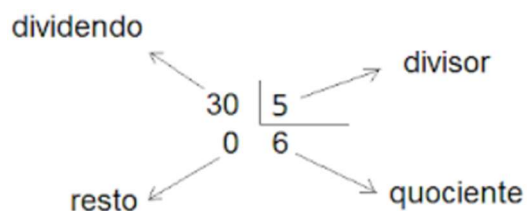
Dessa forma, perceba que o sistema binário não foi inventado para os computadores ou é algo exclusivo à computação, mas sua aplicação no contexto dos computadores, a conversão de binário para decimal realizada pelas máquinas, e a codificação é uma combinação entre arquitetura computacional e lógica matemática.

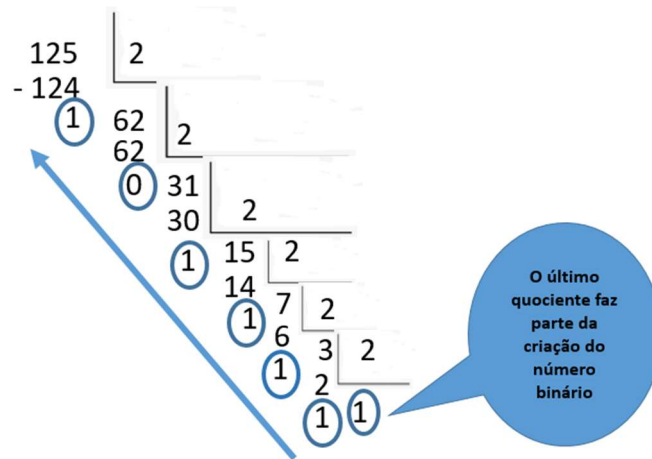
E como funciona o sistema binário, a ideia principal é a junção de 8 bits onde cada bit pode assumir **somente um valor entre dois** (bi-nário) **1 ou 0 - Ligado ou Desligado**.

Vamos montar a nossa tabela binaria

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Vamos converter o 125 em binário?





Vamos somar os bit's que estão ligados ou seja que são 1 inclusive o ultimo quociente

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	1	1	1	1	1	0	1

$$64 + 32 + 16 + 8 + 4 + 1 = 125$$

Legal!!

Mas não tem um jeito mais fácil??

Vamos usar o mesmo 125

Vamos meter a mão na massa.

125 é maior ou igual que 128 → Não → então 0 em 128

125 é maior ou igual que 64 → Sim → 1 no 64 → E subtraímos 64 de 125 → $125 - 64 = 61$

61 é maior ou igual que 32 → Sim → 1 no 32 → Subtraímos 32 de 61 → $61 - 32 = 29$

29 é maior ou igual que 16 → Sim → 1 no 16 → Subtraímos 16 de 29 → $29 - 16 = 13$

13 é maior ou igual que 8 → Sim → 1 no 8 → Subtraímos 8 de 13 → $13 - 8 = 5$

5 é maior ou igual 4 → Sim → 1 no 4 → Subtraímos 4 de 5 → $5 - 4 = 1$

1 é maior ou igual que 2 → Não → 0 no 2

1 é maior ou igual 1 → Sim → 1 no 1 → Subtraímos 1 de 1 → $1 - 1 = 0$

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	1	1	1	1	1	0	1

Exercício

Converte os seguintes números decimais em binário

81 →

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

126 →

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

132 →

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Converte de Binário para decimal

01101010 →

11100011 →

01000001 →