

Notação Posicional

Vamos considerar um número genérico, como 3785, que lemos como três mil, setecentos e oitenta e cinco, esse sistema numérico é o decimal, que os humanos aprendem desde criança, até pelo fato do ser humano ter dez dedos, o que facilitou a aprendizagem da contagem desde o começo da humanidade.

No número 3785, vamos considerar que temos quatro números, o 3 vale três mil, o 7 vale setecentos, o 8 vale oitenta e o 5 é o único que realmente vale 5, dessa forma, podemos escrever o mesmo número assim:

3785:

$$3 \times 1000$$

$$7 \times 100$$

$$8 \times 10$$

$$5 \times 1$$

A base decimal, obviamente, é representada pela base 10, e por isso podemos por potências para definir um número, assim:

3785:

$$3 \times 10^3$$

$$7 \times 10^2$$

$$8 \times 10^1$$

$$5 \times 10^0$$

PS: Lembrando que por regra, qualquer número elevado à zero retorna 1.

Ou seja, a soma do 3785 seria o mesmo que:

3785:

$$3000$$

$$700$$

$$80$$

$$5$$

No caso, o decimal é algo que nós identificamos naturalmente, mas o computador não trabalha só com esse sistema, por isso é importante identificar a base numérica.

Lembrando a matemática básica, os números são trabalhados com unidades, dezenas, centena, milhar, etc., isso é chamado de notação posicional, onde cada algarismo tem sua posição específica e um valor diferente dentro de um número (tipo o 3 em 3785 que representa 3000).

PS: Esse negócio de unidade, dezena, centena, etc., só funciona em decimal.

Pegando outro exemplo em decimal, 2004, separando o milhar, centena, dezena e unidade, dessa forma:

2004:

$$2 = 2 \times 10^3 = 2 \times 1000$$

$$0 = 0 \times 10^2 = 0 \times 100$$

$$0 = 0 \times 10^1 = 0 \times 10$$

$$4 = 4 \times 10^0 = 4 \times 1$$

Lembrando que as potências começam a ser contadas do zero, pela unidade.

Veja outro exemplo com 123:

123:

$$1 = 1 \times 10^2 = 1 \times 100$$

$$2 = 2 \times 10^1 = 2 \times 10$$

$$3 = 3 \times 10^0 = 3 \times 1$$

Assim como sabemos, temos acima da casa de milhar, a dezena de milhar, centena de milhar, milhão, dezena de milhão, centena de milhão, bilhão, etc., quanto maior o número, mais casas decimais, obviamente.

Vamos pegar um exemplo mais longo, por exemplo 134690:

134690:

$$1 = 1 \times 10^5 = 1 \times 100000$$

$$3 = 3 \times 10^4 = 3 \times 10000$$

$$4 = 4 \times 10^3 = 4 \times 1000$$

$$6 = 6 \times 10^2 = 6 \times 100$$

$$9 = 9 \times 10^1 = 9 \times 10$$

$$0 = 0 \times 10^0 = 0 \times 1$$

A potência começa do primeiro algarismo (unidade) com a potência 10^0 , depois a dezena com 10^1 , 10^2 e assim por diante, até infinitas combinações.

Os números em decimal nós reconhecemos naturalmente (como o 3785), mas podemos reconhecê-lo também como $(3785)_{10}$.

Mas até aí, não vimos muita coisa de novo, já que o sistema decimal é o que os humanos entendem, já as máquinas (computador, televisão, celular, etc.) entendem o sistema binário.

Basicamente, os sistemas agem assim:

- **Sistema Decimal:** 0 1 2 3 4 5 6 7 8 9
- **Sistema Binário:** 0 1

No caso, nós teremos que criar uma relação entre os dois sistemas numéricos, para que humanos e máquinas possam interagir entre si.

Vamos contar alguns números em decimal assim:

0 1 2 3 4 5 6 7 8 9

No caso acima, acabou as opções a serem usadas pela unidade, aí colocaremos o 1 na dezena e reiniciar na unidade assim:

10 11 12 13 14 15 16 17 18 19

E assim sucessivamente até chegar em 99, daí colocaremos 1 na centena e reiniciaremos a dezena e unidade (100) até chegar as combinações possíveis (199), aí vamos para 200 até 299 e sucessivamente até 999, e fazemos da mesma forma com 1000 até um número infinito.

Em outras palavras, quando acaba a combinação entre os números de todas as casas, uma a esquerda é adicionada.

Isso daí, pra qualquer um é bem óbvio, mas essa forma também é usada para outros sistemas, vamos contar em binário, que só tem dois dígitos:

No caso, o binário, obviamente, usa a base 2, ficando assim:

0 1

Só nisso acima acabou as possibilidades da primeira casa, tendo que prosseguir colocando o 1 na segunda casa:

10 11

PS: No binário e em outras bases não existem nomenclaturas como unidade, dezena e etc., são identificados apenas como primeira casa, segunda casa e etc. (contato da direita pra esquerda), por isso também não leia 10 como “dez” ou 11 como “onze”, em binário se lê “um-zero” e “um-um”, respectivamente, isso vale pra qualquer número em binário.

Da mesma forma, a medida que for acabando as combinações, colocamos mais uma casa a esquerda, veja abaixo prosseguindo esse raciocínio:

100 101 110 111

E prosseguindo:

1000 1001 1010 1011 1100 1101 1110 1111

Da mesma forma:

10000 10001 10010 10011 10100 10101 10110 10111 11000 11001 11010 11011 11100 11101 11110 11111

E assim sucessivamente.

Da mesma forma que fizemos com os decimais, podemos usar a base 2 para converter binários em decimais, assim:

1011:

$$1 = 1 \times 2^3 = 1 \times 8 = 8$$

$$0 = 0 \times 2^2 = 0 \times 4 = 0$$

$$1 = 1 \times 2^1 = 1 \times 2 = 2$$

$$1 = 1 \times 2^0 = 1 \times 1 = 1$$

$$8 + 0 + 2 + 1 = 11$$

Dessa forma é fácil converter binário em decimal, sempre lembrando do resultado das potências pra cada casa, a posição onde encontra o 1 é somada, a posição onde encontra 0 é ignorada, e a soma de todos retorna o decimal.

Da mesma forma:

1001:

$$1 = 1 \times 2^3 = 1 \times 8 = 8$$

$$0 = 0 \times 2^2 = 0 \times 4 = 0$$

$$0 = 0 \times 2^1 = 0 \times 2 = 0$$

$$1 = 1 \times 2^0 = 1 \times 1 = 1$$

$$8 + 0 + 0 + 1 = 9$$

PS: Os números na base 2 são representados, por exemplo, como (1001)₂.

Números Binários

Como sabemos, os aparelhos eletrônicos usam sistema binário, já que eles trabalham com pulsos elétricos, que justamente é representado por binário (que trabalha por padrões como ligado/desligado), e eles são convertidos para que a gente possa entender.

Ramos como hardware e redes necessitam muito de uso de sistemas numéricos diferentes, como o binário e o hexadecimal.

Como visto, existe uma ligação entre os sistemas numéricos decimal e binário, até porque a matemática é uma ciência exata.

Veja um exemplo de conversão de binário para decimal:

1101:

$$1 = 1 \times 2^3 = 1 \times 8 = 8$$

$$1 = 1 \times 2^2 = 1 \times 4 = 4$$

$$0 = 0 \times 2^1 = 0 \times 2 = 0$$

$$1 = 1 \times 2^0 = 1 \times 1 = 1$$

$$8 + 4 + 0 + 1 = 13$$

Isso já vimos anteriormente, é só pra lembrar.

Mas temos um macete bem simples para converter binário em decimal facilmente, é só lembrar dos resultados das potências na ordem das casas numéricas, 1, 2, 4, 8, 16, 32, etc., por exemplo:

16	8	4	2	1
----	---	---	---	---

1	0	0	1	1
---	---	---	---	---

No caso acima, podemos simplesmente ignorar os resultados de potências que tem 0 e somar os que tem apenas 1, dessa forma, pro mesmo número acima:

$$16 + 0 + 0 + 2 + 1 = 19$$

Dessa forma, o número 10011 (não leia dez mil e onze, é um zero zero um um) convertido para decimal é igual a 19.

Já o contrário, ou seja, converter em decimal pra binário (por exemplo, 0 13 na base 10), é fazendo divisão inteira, dessa forma:

$$\begin{array}{r}
 13 \overline{) 2} \\
 1 \quad 6 \overline{) 2} \\
 0 \quad 3 \overline{) 2} \\
 1 \quad 1 \overline{) 2} \\
 1 \quad 0
 \end{array}$$

$$13 = 1101$$

Ou seja, vamos dividindo até não dar mais (ou seja, o resultado ser zero), no caso da divisão inteira, 1 dividido por 2 o resultado é zero e sobra 1 (até porque nesse caso não dá pra dividir um por 2), e 0 dividido por 2 é 0 e sobra 0. Essa regra existe sempre quando o numerador é menor que o denominador.

Aí, pegamos o número de baixo pra cima, o primeiro resto (mais abaixo) é o primeiro dígito (à esquerda), e o último resto (na casa mais acima), é o último dígito (à direita). No caso, o 13 em binário é 1101.

Temos também um macete para converter um número decimal em binário, usando cada casa com a potência numérica em cima também:

8	4	2	1
1	1	0	1

No caso, basta somar os resultados das potências acima até ver se elas retornam o número, no caso, $8 + 4 + 1$.

Aí basta colocar o 1 onde tem os números das somas, e 0 onde não tem.

Veja outro exemplo com 40 na base 10:

32	16	8	4	2	1
1	0	1	0	0	0

No caso, a soma de $32 + 8$ retornou 40, que no binário é 101000.

Para provar, fazemos a reconversão de binário pra decimal:

101000:

$$1 = 1 \times 2^5 = 1 \times 32 = 32$$

$$0 = 0 \times 2^4 = 0 \times 16 = 0$$

$$1 = 1 \times 2^3 = 1 \times 8 = 8$$

$$0 = 0 \times 2^2 = 0 \times 4 = 0$$

$$0 = 0 \times 2^1 = 0 \times 2 = 0$$

$$0 = 0 \times 2^0 = 0 \times 1 = 0$$

$$32 + 0 + 8 + 0 + 0 + 0 = 40$$

Números Octais e Hexadecimais

Como visto até agora, dois sistemas numéricos usados são o decimal, usado pelos humanos, e o binário, usado pelas máquinas, mas ainda tem outros sistemas utilizados na área de TI, no caso, veremos o sistema octal e hexadecimal.

Números de IPv6 e MAC, ou hashes, são usados em hexadecimal, por exemplo. Códigos de erros em jogos ou de memória, ou permissões em Linux, são usados em octal.

Basicamente, o sistema octal é de 0 a 7 (eliminado o oito, assim como o binário elimina o dois e o decimal só vai até nove). O sistema hexadecimal vai de 0 a 9 e depois tem seis letras de A a F (que não são lidos como letras do alfabeto. Veja abaixo os algarismos dessas bases:

- **Sistema Decimal:** 0 1 2 3 4 5 6 7 8 9.
- **Sistema Binário:** 0 1.
- **Sistema Octal:** 0 1 2 3 4 5 6 7.
- **Sistema Hexadecimal:** 0 1 2 3 4 5 6 7 8 9 A B C D E F.

PS: Nos números octais e hexadecimais, assim como os binários, as casas também não são contadas por unidades, dezenas e etc, e não se lê coisas como “dez mil, cento e onze”.

Vamos relembrar o número 2004 em decimal:

2004:

$$2 = 2 \times 10^3 = 2 \times 1000$$

$$0 = 0 \times 10^2 = 0 \times 100$$

$$0 = 0 \times 10^1 = 0 \times 10$$

$$4 = 4 \times 10^0 = 4 \times 1$$

Isso acima é nada mais do que a notação posicional.

Da mesma forma, a notação é usada para qualquer base, lembrando do binário:

1101:

$$1 = 1 \times 2^3 = 1 \times 8 = 8$$

$$1 = 1 \times 2^2 = 1 \times 4 = 4$$

$$0 = 0 \times 2^1 = 0 \times 2 = 0$$

$$1 = 1 \times 2^0 = 1 \times 1 = 1$$

$$8 + 4 + 0 + 1 = 13$$

Vamos ver por exemplo, a notação posicional em octal (base 8), no caso, o 371:

371:

$$3 = 3 \times 8^2 = 3 \times 64 = 192$$

$$7 = 7 \times 8^1 = 7 \times 8 = 56$$

$$1 = 1 \times 8^0 = 1 \times 1 = 1$$

$$192 + 56 + 1 = 249$$

No caso acima, 371 na base 8 (leia três sete um em octal) é o mesmo que 249 em hexadecimal (esse pode ser lido duzentos e quarenta e nove). A base 8 pode ser lida como, por exemplo, $(371)_8$.

Agora, no sistema hexadecimal (base 16), é bom lembrar primeiramente que os caracteres A, B, C, D, E e F são o mesmo que 10, 11, 12, 13, 14 e 15 em decimal.

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Vamos ver um exemplo de um número em hexadecimal, por exemplo, o 1FA:

1FA:

$$A = 10$$

$$F = 15$$

$$1 = 1 \times 16^2 = 1 \times 256 = 256$$

$$F = 15 \times 16^1 = 15 \times 16 = 240$$

$$A = 10 \times 16^0 = 10 \times 1 = 10$$

$$256 + 240 + 10 = 506$$

No caso acima, 1FA na base 16 (leia um efe a) é o mesmo que 506 em decimal (esse pode ser lido quinhentos e seis). Os números em hexadecimal podem ser lidos, como por exemplo, $(1FA)_{16}$.

Veja outro exemplo, com 1257 na base 8:

1257:

$$1 = 1 \times 8^3 = 1 \times 512 = 512$$

$$2 = 2 \times 8^2 = 2 \times 64 = 128$$

$$5 = 5 \times 8^1 = 5 \times 8 = 40$$

$$7 = 7 \times 8^0 = 7 \times 1 = 7$$

$$512 + 128 + 40 + 7 = 687$$

No caso acima, o 1257 em octal é o mesmo que 687 em decimal.

Veja também outro exemplo, com 13C2 na base 16:

13C2:

$$C = 12$$

$$1 = 1 \times 16^3 = 1 \times 4096 = 4096$$

$$3 = 3 \times 16^2 = 3 \times 256 = 768$$

$$C = 12 \times 16^1 = 12 \times 16 = 192$$

$$2 = 2 \times 16^0 = 2 \times 1 = 2$$

$$4096 + 768 + 192 + 2 = 5058$$

No caso acima, o 13C2 em hexadecimal é o mesmo que 5058 em decimal.

Dessa mesma forma, é possível fazer conversões de números de qualquer base, inclusive de bases não citadas como base 7 ou base 3.

Como visto anteriormente, podíamos converter decimal em binário fazendo divisões inteiras, dessa forma:

$$\begin{array}{r} 13 \overline{) 2} \\ 1 \quad 6 \overline{) 2} \\ 0 \quad 3 \overline{) 2} \\ 1 \quad 1 \overline{) 2} \\ 1 \quad 0 \end{array}$$

$$13 = 1101$$

Da mesma forma, podemos converter decimal em octal, como podemos ver o número 177 em decimal:

$$\begin{array}{r} 177 \overline{) 8} \\ 1 \quad 22 \overline{) 8} \\ 6 \quad 2 \overline{) 8} \\ 2 \quad 0 \end{array}$$

$$177 = 261$$

Da mesma forma, vamos dividindo até não dar mais (ou seja, o resultado ser zero), no caso da divisão inteira, quando o numerador é menor que o denominador, o resultado é zero e o resto é igual ao numerador.

Aí, da mesma forma, pegamos o número de baixo pra cima, o primeiro resto (abaixo) é a primeira casa (à esquerda), e o último resto (na casa mais acima), é o último dígito (à direita). No caso, o 177 em octal é 261.

Para vermos que está certo, vamos reconverter o octal em decimal:

261:

$$2 = 2 \times 8^2 = 2 \times 64 = 128$$

$$6 = 6 \times 8^1 = 6 \times 8 = 48$$

$$1 = 1 \times 8^0 = 1 \times 1 = 1$$

$$128 + 48 + 1 = 177$$

A forma para converter decimal em hexadecimal é a mesma, como podemos ver com o número 685 em decimal:

$$\begin{array}{r} 685 \mid \underline{16} \\ 13 \quad 42 \mid \underline{16} \\ 10 \quad 2 \mid \underline{16} \\ 2 \quad 0 \end{array}$$

$$13 = D$$

$$10 = A$$

$$685 = 2AD$$

Aí, da mesma forma, pegamos o número de baixo pra cima, o primeiro resto (abaixo) é a primeira casa (à esquerda), e o último resto (na casa mais acima), é o último dígito (à direita), e tudo que retornar de 10 e 15 trocamos por A a F. No caso, o 685 em hexadecimal é 2AD.

Para vermos que está certo, vamos reconverter o hexadecimal em decimal:

$$2AD:$$

$$A = 10$$

$$D = 13$$

$$2 = 2 \times 16^2 = 2 \times 256 = 512$$

$$A = 10 \times 16^1 = 10 \times 16 = 160$$

$$D = 13 \times 16^0 = 13 \times 1 = 13$$

$$512 + 160 + 13 = 685$$

Podemos também converter outras bases, por exemplo, para hexadecimal em binário, vamos converter o 1E na base 16.

Primeiro, converteremos o 1E para decimal, assim:

$$1E:$$

$$E = 14$$

$$1 = 1 \times 16^1 = 1 \times 16 = 16$$

$$E = 14 \times 16^0 = 14 \times 1 = 14$$

$$16 + 14 = 30$$

E depois, converteremos o resultado decimal para binário:

$$30 \mid \underline{2}$$

$$\begin{array}{r}
 0 \ 15 \mid \underline{2} \\
 1 \ 7 \mid \underline{2} \\
 1 \ 3 \mid \underline{2} \\
 1 \ 1 \mid \underline{2} \\
 1 \ 0
 \end{array}$$

$$30 = 11110$$

Daí é só organizar, e veremos que 1E na base 16 é igual a 11110 na base 2. Dessa forma podemos ver que existe relação entre os sistemas numéricos.

No caso, o sistema decimal serviu de ponte para conversões, isso pode ser usado para qualquer conversão, como de octal para hexadecimal, por exemplo.

Relação Entre Sistemas Numéricos

Como visto anteriormente, quando precisamos converter entre sistemas (como o octal e hexadecimal), usamos o decimal como intermediário para conversões.

Vamos ver um exemplo, onde jogaremos o número 27 em octal para o binário:

27:

$$\begin{aligned}
 2 &= 2 \times 8^1 = 2 \times 8 = 16 \\
 7 &= 7 \times 8^0 = 7 \times 1 = 7
 \end{aligned}$$

$$16 + 7 = 23$$

No caso acima o 27 na base 8 retornou 23 na base 10, e para converter o decimal em binário:

$$\begin{array}{r}
 23 \mid \underline{2} \\
 1 \ 11 \mid \underline{2} \\
 1 \ 5 \mid \underline{2} \\
 1 \ 2 \mid \underline{2} \\
 0 \ 1 \mid \underline{2} \\
 1 \ 0
 \end{array}$$

E escrevendo de trás pra frente, o 27 em octal é 10111 em binário.

Mas temos um macete muito mais fácil para converter octal em binário. Isso ocorre porque a cada três dígitos do binário representa um dígito do octal (7 em octal é 111 em binário), já que 2 (base binário) elevado à 3 é 8 (base octal).

Sabendo isso, podemos dividir um número octal em três dígitos, veja um exemplo com o mesmo 27 em octal:

$$\begin{aligned}
 2 &= \underline{\quad} \\
 7 &= \underline{\quad}
 \end{aligned}$$

E cada uma dessas casas, colocamos as potências, igual quando convertemos decimal em binário.

4	2	1
---	---	---

No caso, basta fazer a soma e onde tiver os números somados, coloque 1, e onde não tiver coloque 0, assim:

$$2 = 010$$

$$7 = 111$$

Aí é só juntar tudo e veremos que 27 em octal é 10111 (o zero à esquerda pode ser descartado no binário e em outros sistemas numéricos).

E para converter binário para octal, seguimos a mesma lógica da potência de 2^3 . Dividimos o binário em pequenos grupos de três.

Veja a conversão de 1000101 em binário para octal:

$$001\ 000\ 101$$

PS: Caso não seja possível completar o grupo com três, complete com zeros à esquerda.

Seguindo a mesma lógica, vamos usar os resultados das potências (4, 2, 1) e somar tudo que tiver abaixo de um, ou seja:

$$001\ 000\ 101 - \text{Binário puro}$$

$$001\ 000\ 401 - \text{Cada dígito das potências}$$

$$1\ 0\ 5 - \text{Octais retornados}$$

Dessa forma, 1000101 na base 2 é o mesmo que 105 na base octal.

PS: Lembre-se que zeros no meio são representados.

Agora vamos ver um exemplo, onde jogaremos o número 1B em hexadecimal para o binário:

$$1B:$$

$$B = 11$$

$$1 = 1 \times 16^1 = 1 \times 16 = 16$$

$$B = 11 \times 16^0 = 11 \times 1 = 11$$

$$16 + 11 = 27$$

No caso acima, o 1B na base 16 retornou 27 na base 10, e para converter o decimal em binário:

$$\begin{array}{r} 27 \overline{) 2} \\ 1 \ 13 \overline{) 2} \\ \quad 1 \ 6 \overline{) 2} \\ \qquad 0 \ 3 \overline{) 2} \\ \qquad \quad 1 \ 1 \overline{) 2} \\ \qquad \qquad 1 \ 0 \end{array}$$

E escrevendo de trás pra frente, o 1B em hexadecimal é 11011 em binário.

Da mesma forma, temos um macete muito mais fácil para converter hexadecimal em binário. Isso ocorre porque a cada quatro dígitos do binário representa um dígito do octal (F em hexadecimal é 1111 em binário), já que 2 (base binária) elevado à 4 é 16 (base hexadecimal).

Sabendo isso, podemos dividir um número hexadecimal em quatro dígitos, veja um exemplo com o mesmo 1B em hexadecimal:

$$B = 11$$

$$1 = \underline{\quad}$$

$$11 = \underline{\quad}$$

E cada uma dessas casas, colocamos as potências, igual quando convertemos decimal em binário.

8	4	2	1
---	---	---	---

No caso, basta fazer a soma e onde tiver os números somados, coloque 1, e onde não tiver coloque 0, assim:

$$B = 11$$

$$1 = 0001$$

$$11 = 1011$$

Aí é só juntar tudo e veremos que 1B em hexadecimal é 11011 (o zero à esquerda pode ser descartado no binário e em outros sistemas numéricos).

E para converter binário para hexadecimal, seguimos a mesma lógica da potência de 2^4 . Dividimos o binário em pequenos grupos de quatro.

Veja a conversão de 1011101 em binário para hexadecimal:

$$0101 \ 1101$$

PS: Caso não seja possível completar o grupo com quatro, complete com zeros à esquerda.

Seguindo a mesma lógica, vamos usar o resultado das potências (8, 4, 2, 1) e somar tudo que tiver abaixo de um, ou seja:

$$0101 \ 1101 - \text{Binário puro}$$

$$0401 \ 8401 - \text{Cada dígito das potências}$$

$$5 \quad 13(D) - \text{Hexadecimais retornados}$$

Dessa forma, 1011101 na base 2 é o mesmo que 5D na base hexadecimal.

PS: Lembre-se que zeros no meio também são representados, e se o resultado for entre 10 e 15, deverá ser substituído pelos caracteres entre A e F.

Aritmética nas Bases 2, 8 e 16

A aritmética de números nas bases 2, 8 e 16 funciona da mesma maneira que na base 10, obedecendo as opções de números.

Veja dois exemplos em binário:

$$\begin{array}{r} 1011 \\ +101 \\ \hline 10000 \end{array}$$

Só lembrando que $1 + 1$ é igual a 10, e $10 + 1$ é igual a 11.

Pra subtrair a mesma regra:

$$\begin{array}{r} 1010 \\ -111 \\ \hline 11 \end{array}$$

Nesse caso precisamos lembrar que $10 - 1$ é igual a 1, e $11 - 1$ é igual a 10. A regra de “emprestar” números quando for subtrair 1 de 0, permanece.

No caso de octais a mesma coisa, por exemplo:

$$\begin{array}{r} 176 \\ +43 \\ \hline 241 \end{array}$$

Só precisamos lembrar que na soma, caso dê 8, converta para 10, 9 para 11, 10 para 12 e assim por diante.

Na subtração, a mesma coisa:

$$\begin{array}{r} 542 \\ -35 \\ \hline 505 \end{array}$$

Nesse caso, podemos converter os octais para decimal (no caso 12 vira 10 e fazer a subtração, e converter o resultado para octal. As regras de “emprestar” permanecem.

No caso de hexadecimal também, por exemplo:

$$\begin{array}{r} 4E3 \\ +A5 \\ \hline 588 \end{array}$$

Só precisamos lembrar que na soma, caso dê 10, converta para A, 11 para B, 12 para C e assim por diante.

Na subtração, a mesma coisa:

$$\begin{array}{r} A13 \\ -5E \\ \hline 9B5 \end{array}$$

Nesse caso, podemos converter os hexadecimais para decimal (no caso (E vira 14) e fazer a subtração, e converter o resultado para hexadecimal. As regras de “emprestar” permanecem.

PS: Tanto na base octal quanto na hexadecimal, é interessante converter os algarismos em binário, octal ou hexadecimal para decimal, fazer a conta e reconverter para a base especificada.

Revisão de Tabuada e Potências

Para facilitar os cálculos entre bases 2, 8, 10 e 16, é bom sabermos das tabuadas desses números. Veja abaixo elas:

Tabuada de 2:

$$\begin{aligned} 2 \times 1 &= 2 \\ 2 \times 2 &= 4 \\ 2 \times 3 &= 6 \\ 2 \times 4 &= 8 \\ 2 \times 5 &= 10 \\ 2 \times 6 &= 12 \\ 2 \times 7 &= 14 \\ 2 \times 8 &= 16 \\ 2 \times 9 &= 18 \\ 2 \times 10 &= 20 \end{aligned}$$

Tabuada de 8:

$$\begin{aligned} 8 \times 1 &= 8 \\ 8 \times 2 &= 16 \\ 8 \times 3 &= 24 \\ 8 \times 4 &= 32 \\ 8 \times 5 &= 40 \\ 8 \times 6 &= 48 \\ 8 \times 7 &= 56 \\ 8 \times 8 &= 64 \\ 8 \times 9 &= 72 \\ 8 \times 10 &= 80 \end{aligned}$$

Tabuada de 10:

$$\begin{aligned} 10 \times 1 &= 10 \\ 10 \times 2 &= 20 \\ 10 \times 3 &= 30 \\ 10 \times 4 &= 40 \\ 10 \times 5 &= 50 \\ 10 \times 6 &= 60 \\ 10 \times 7 &= 70 \\ 10 \times 8 &= 80 \\ 10 \times 9 &= 90 \end{aligned}$$

$$10 \times 10 = 100$$

Tabuada de 16:

$$16 \times 1 = 16$$

$$16 \times 2 = 32$$

$$16 \times 3 = 48$$

$$16 \times 4 = 64$$

$$16 \times 5 = 80$$

$$16 \times 6 = 96$$

$$16 \times 7 = 112$$

$$16 \times 8 = 128$$

$$16 \times 9 = 144$$

$$16 \times 10 = 160$$

PS: Na matemática, a tabuada é de 0 a 10, mas colocamos as multiplicações de 16 para facilitar as conversões com os números hexadecimais.

Veja também as potências desses números.

Potências de 2:

Potência	Resultado
2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64
2^7	128
2^8	256
2^9	512
2^{10}	1024
2^{11}	2048
2^{12}	4096
2^{13}	8192
2^{14}	16384
2^{15}	32768
2^{16}	65536
2^{17}	131072
2^{18}	262144
2^{19}	524288

2^{20}	1048576
----------	---------

Potências de 8:

Potência	Resultado
8^0	1
8^1	8
8^2	64
8^3	512
8^4	4096
8^5	32768
8^6	262144
8^7	2097152
8^8	1677216

Potências de 10:

Potência	Resultado
10^0	1
10^1	10
10^2	100
10^3	1000
10^4	10000
10^5	100000

Potências de 16:

Potência	Resultado
16^0	1
16^1	16
16^2	256
16^3	4096
16^4	65536
16^5	1048576
16^6	16777216
16^7	268435456