

# AULA 3 – SOMA, SUBTRAÇÃO, DIVISÃO E MULTIPLICAÇÃO

## OBJETIVO DA AULA

Efetuar operações aritméticas nas bases binária, octal e hexadecimal.

## APRESENTAÇÃO

Na aula anterior conhecemos as bases binária, octal e hexadecimal. Além disso, aprendemos como converter números entre elas.

Agora que já sabemos como funcionam as bases binária, octal e hexadecimal e como converter números entre elas, está na hora de aprendermos a somar, subtrair, multiplicar e dividir nelas.

As operações funcionam da mesma forma que na base decimal. Então, é hora de começarmos.

## 1. SOMA EM BINÁRIO

Para começarmos, vamos lembrar que:

$$0 + 0 = 0.$$

$$0 + 1 = 1.$$

$$1 + 0 = 1.$$

$$1 + 1 = 10 (= 2 \text{ em binário} - \text{ vamos usar o "vai 1"}).$$

$$1 + 1 + 1 = 11 (= 3 \text{ em binário} - \text{ e vamos usar o "vai 1" também}).$$

Como exemplo, vamos somar os números  $(1101)_2$  e  $(1011)_2$ .

$$\begin{array}{r}
 1101 \\
 + 1011 \\
 \hline
 0 \text{ Vai 1}
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 + 1011 \\
 \hline
 00 \text{ Vai 1}
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 + 1011 \\
 \hline
 000 \text{ Vai 1}
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 + 1011 \\
 \hline
 1000 \text{ Vai 1}
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 + 1011 \\
 \hline
 11000
 \end{array}$$

Repare que o processo é o mesmo utilizado nas somas em decimal. E você verá que em octal e hexadecimal será da mesma forma.

## 1.1. SOMA EM OCTAL

Aqui precisamos lembrar que, como não usamos os algarismos 8 e 9, sempre que o resultado da soma igualar ou passar de 7, teremos o *vai 1*.

O algoritmo é simples:

1º caso: o resultado da soma está entre 0 e 7 – simplesmente o escrevemos, sem precisar o *vai 1*.

2º caso: o resultado iguala ou ultrapassa 8 – então executaremos os seguintes passos:

- Somamos 1 à próxima coluna (*vai 1*);
- Na coluna atual colocamos o quanto ultrapassou a base (no caso, 8).

Observe o exemplo abaixo, somando  $(1563 + 1275)_8$ .

			1							1	1							1	1							1	1				
		1	5	6	3					1	5	6	3					1	5	6	3					1	5	6	3		
	+	1	2	7	5				+	1	2	7	5				+	1	2	7	5				+	1	2	7	5		
					0							6	0							0	6	0						3	0	6	0
(a)								(b)								(c)								(d)							

a)  $3 + 5 = 8 \rightarrow$  colocamos o resultado igual a 0 (zero), pois devemos subtrair 8 (resultado maior do que 7) por 8 (a base octal). Então fica:  $8 - 8 = 0$ . Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda.

b)  $1 + 6 + 7 = 14 \rightarrow$  colocamos o resultado igual a 6, pois devemos subtrair 14 (resultado maior do que 7) por 8 (a base octal). Então fica:  $14 - 8 = 6$ . Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda.

c)  $1 + 5 + 2 = 8 \rightarrow$  colocamos o resultado igual a 0 (zero), pois devemos subtrair 8 (resultado maior do que 7) por 8 (a base octal). Então fica:  $8 - 8 = 0$ . Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda.

d)  $1 + 1 + 1 = 3 \rightarrow$  colocamos o resultado igual a 3 e, como está entre 0 e 7, não precisamos adicionar nada na próxima coluna.

## 1.2. SOMA EM HEXADECIMAL

Aqui usaremos o mesmo raciocínio. Mas lembre-se: como estamos trabalhando na base decimal, tudo o que fizemos em relação a 8 na base octal faremos em relação ao 16 na base hexadecimal.

Assim, quando a soma ficar entre 0 e 15, usaremos o algarismo (ou letra) correspondente, sem precisar adicionar 1 à próxima coluna. E quando igualar ou passar de 16 utilizaremos o mesmo raciocínio do octal.

Como exemplo, vamos efetuar  $(AF97 + BC59)_{16}$ .

			1	
	A	F	9	7
+	B	C	5	9
			F	0

(a)

	1		1	
	A	F	9	7
+	B	C	5	9
		B	F	0

(b)

	1	1		1	
		A	F	9	7
+	B	C	5	9	
	6	B	F	0	

(c)

	1	1			1	
		A	F	9	7	
+	B	C	5	9		
	1	6	B	F	0	

(d)

a)  $7 + 9 = 16 \rightarrow$  colocamos o resultado igual a 0 (zero), pois devemos subtrair 16 (resultado maior do que 15) por 16 (a base hexadecimal). Então fica:  $16 - 16 = 0$ . Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda.

b)  $1 + 9 + 5 = 15 \rightarrow$  como 15 é menor do que 16, colocamos a letra correspondente ao 15, a letra F.

c)  $15(F) + 12(C) = 27 \rightarrow$  colocamos o resultado igual a B (11), pois devemos subtrair 27 (resultado maior do que 15) por 16 (a base hexadecimal). Então fica:  $27 - 16 = 11$  (B). Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda.

d)  $1 + 10(A) + 11(B) = 22 \rightarrow$  colocamos o resultado igual a 6, pois devemos subtrair 22 (resultado maior do que 15) por 16 (a base hexadecimal). Então fica:  $22 - 16 = 6$ . Devemos adicionar 1 a próxima coluna a esquerda, pois a regra informa que ao ultrapassar o valor, subtraímos, inserimos o resultado e adicionamos 1 a próxima coluna a esquerda. Neste caso, ao subir o 1, consequentemente faremos a  $1 + 0$  (nada, vazio da coluna) e será igual a 1. Resultado: 16BF0

## 2. SUBTRAÇÃO EM BINÁRIO

Agora vamos usar, como em decimal, o recurso de “pedir emprestado”. Lembrando uma regra simples: **quem empresta perde um e quem ganha emprestado ganha o valor da base.** Isso valerá para todas as bases com que trabalhamos aqui.

Vamos efetuar, como exemplo  $(110001 - 101111)_2$ .

$$\begin{array}{r} 110001 \\ - 101111 \\ \hline \end{array}$$

Subtração trivial

$$\begin{array}{r} 011100 \\ 110001 \\ - 101111 \\ \hline 10 \end{array}$$

Como foi necessário “pedir emprestado”, o primeiro 1 encontrado à esquerda vira zero e haverá uma sequência de empréstimos até a coluna em que estamos trabalhando. Nesse caso o 0 ganha 2 (10). Teremos então  $(10 - 1)_2 = 1$

$$\begin{array}{r} 011100 \\ 110001 \\ - 101111 \\ \hline 010 \end{array}$$

A partir daqui as subtrações são triviais e só teremos zeros à esquerda

### 2.1. SUBTRAÇÃO EM OCTAL

Aqui também precisaremos do recurso de “pedir emprestado”, lembrando de que quem emprestar perderá 1 e quem receber ganhará 8.

Vamos ao exemplo efetuando  $(7214 - 1636)_8$ .

$$\begin{array}{r} 7214 \\ - 1636 \\ \hline \end{array}$$

Como 4 é menor do que 7, terá que pedir emprestado ao 1

$$\begin{array}{r} 12 \\ 7204 \\ - 1636 \\ \hline 6 \end{array}$$

Ao emprestar, o 1 virou 0 e o 4, recebendo 8, virou 12. Então,  $12 - 6 = 6$

$$\begin{array}{r} 8 \\ 7104 \\ - 1636 \\ \hline 56 \end{array}$$

Como 0 é menor do que 3, terá que pedir emprestado ao 2, que vira 1 e o 0, ao receber emprestado, vira 8. Finalmente,  $8 - 3 = 5$

$$\begin{array}{r} 9 \\ 6104 \\ - 1636 \\ \hline 5356 \end{array}$$

Como 1 é menor do que 6, terá que pedir emprestado ao 7, que vira 6 e o 1, ao receber emprestado, vira 9. Finalmente,  $9 - 6 = 3$

Finalizando,  $6 - 1 = 5$

## 2.2. SUBTRAÇÃO EM HEXADECIMAL

Aqui executaremos o mesmo algoritmo, sempre lembrando que os valores entre 10 e 15 serão substituídos pelas letras e que, na regra do pedir emprestado, aquele que receber o empréstimo ganhará 16.

Vamos efetuar  $(A741 - 8BED)_{16}$ .

$\begin{array}{r} A741 \\ - 8BED \\ \hline \end{array}$	Como 1 é menor do que D (13), terá que pedir emprestado ao 4
$\begin{array}{r} A7\overset{17}{3}1 \\ - 8BED \\ \hline \end{array}$	Ao emprestar, o 4 virou 3 e o 1, recebendo 16, virou 17. Então, $17 - 13 = 4$
$\begin{array}{r} A6\overset{19}{3}1 \\ - 8BED \\ \hline \end{array}$	Como 3 é menor do que E (14), terá que pedir emprestado ao 7, que vira 6 e o 3, ao receber emprestado, vira 19. Finalmente, $19 - 14 = 5$
$\begin{array}{r} \overset{22}{9}6\overset{11}{3}1 \\ - 8BED \\ \hline \end{array}$	Como 6 é menor do que B (11), terá que pedir emprestado ao A, que vira 9 e o 6, ao receber emprestado, vira 22. Finalmente, $22 - 11 = 11$ (B)
$\begin{array}{r} 1B56 \\ \hline \end{array}$	Finalizando, $9 - 8 = 1$

## 3. MULTIPLICAÇÃO EM BINÁRIO

Aqui também vamos utilizar o mesmo algoritmo que usamos nas multiplicações em decimal.

Lembrando que  $0 \times 0 = 0$ ,  $0 \times 1 = 0$  e  $1 \times 1 = 1$ .

Como exemplo, vamos efetuar  $(1101 \times 101)_2$ .

$\begin{array}{r} 1101 \\ \times 101 \\ \hline 1101 \\ 00000 \\ + 110100 \\ \hline 1000001 \end{array}$	O primeiro passo é multiplicar cada algarismo do número de baixo pelo número de cima.
	O segundo passo é somar as parcelas.

### 3.1. MULTIPLICAÇÃO EM OCTAL

Para facilitar essa operação, vamos utilizar uma tabela com uma “tabuada” em octal.

x	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7

X	0	1	2	3	4	5	6	7
2	0	2	4	6	10	12	14	16
3	0	3	6	11	14	17	22	25
4	0	4	10	14	20	24	30	34
5	0	5	12	17	24	31	36	43
6	0	6	14	22	30	36	44	52
7	0	7	16	25	34	43	52	61

Com ajuda dessa tabela, vamos efetuar  $(32 \times 45)_8$ .

$$\begin{array}{r}
 \overset{1}{45} \\
 \times 32 \\
 \hline
 2
 \end{array}$$

Usando nossa tabela, veremos que  $2 \times 5 = 12$ . Assim colocamos o 2 embaixo e subimos o 1

$$\begin{array}{r}
 \overset{1}{45} \\
 \times 32 \\
 \hline
 112
 \end{array}$$

$2 \times 4 = 10$ ; somando com o 1 que subiu, teremos 11.

$$\begin{array}{r}
 \overset{1}{45} \\
 \times 32 \\
 \hline
 112 \\
 157
 \end{array}$$

Agora efetuamos a multiplicação por 3.

$$\begin{array}{r}
 45 \\
 \times 32 \\
 \hline
 112 \\
 + 157 \\
 \hline
 1702
 \end{array}$$

Finalmente, efetuamos a soma.

### 3.2. MULTIPLICAÇÃO EM HEXADECIMAL

Da mesma forma que fizemos com a base octal, vamos usar uma tabela para nos auxiliar.

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	2	4	6	8	A	C	E	10	13	14	16	18	1A	1C	1E
3	0	3	6	9	C	F	12	15	18	1B	1E	21	24	27	2A	2D
4	0	4	8	C	10	14	18	1C	20	24	28	2C	30	34	38	3C
5	0	5	A	F	14	19	1E	23	28	2D	32	37	3C	41	46	4B
6	0	6	C	12	18	1E	24	2A	30	36	3C	42	48	4E	54	5A
7	0	7	E	15	1C	23	2A	31	38	3F	46	4D	54	5B	62	69
8	0	8	10	18	20	28	30	38	40	48	50	58	60	68	70	78
9	0	9	13	1B	24	2D	36	3F	48	51	5A	63	6C	75	7E	87
A	0	A	14	1E	28	32	3C	46	50	5A	64	6E	78	82	8C	96

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
B	0	B	16	21	2C	37	42	4D	58	63	6E	69	84	8F	9A	A5
C	0	C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4
D	0	D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3
E	0	E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2
F	0	F	1E	2D	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1

Como exemplo, vamos efetuar  $(3F \times D9)_{16}$ .

$$\begin{array}{r}
 \begin{array}{r}
 \overset{8}{D9} \\
 \times \overset{8}{3F} \\
 \hline
 7
 \end{array}
 \quad
 \begin{array}{l}
 \text{Usando nossa tabela, veremos que } F \times 9 = 87. \text{ Assim} \\
 \text{colocamos o 7 embaixo e subimos o 8}
 \end{array} \\
 \begin{array}{r}
 \overset{8}{D9} \\
 \times \overset{8}{3F} \\
 \hline
 CB7
 \end{array}
 \quad
 \begin{array}{l}
 F \times D = C3; \text{ somando com o 8 que subiu, teremos CB.}
 \end{array} \\
 \begin{array}{r}
 \overset{1}{D9} \\
 \times \overset{3}{3F} \\
 \hline
 CB7 \\
 28B
 \end{array}
 \quad
 \begin{array}{l}
 \text{Agora efetuamos a multiplicação por 3.}
 \end{array} \\
 \begin{array}{r}
 D9 \\
 \times 3F \\
 \hline
 CB7 \\
 +28B \\
 \hline
 3567
 \end{array}
 \quad
 \begin{array}{l}
 \text{Finalmente, efetuamos a soma.}
 \end{array}
 \end{array}$$

## 4. DIVISÃO EM BINÁRIO

A operação de divisão em binário segue o mesmo procedimento da divisão em binário, com a diferença de que quando o valor do dividendo é menor do que o divisor, o quociente recebe 0. Se o dividendo foi maior ou igual ao divisor, o quociente recebe 1.

Para entender melhor, vamos a um exemplo, efetuando  $(111010 \div 11)_2$ .

$$\begin{array}{r} 111010 \overline{) 11} \\ 0 \quad 1 \end{array}$$

O dividendo em questão é igual ao divisor. Assim, colocamos 1 no quociente e 0 no resto.

$$\begin{array}{r} 111010 \overline{) 11} \\ 01 \quad 10 \end{array}$$

Agora o dividendo é menor do que o divisor. Então colocamos o 0 no quociente e "baixamos" mais um número.

$$\begin{array}{r} 111010 \overline{) 11} \\ 010 \quad 100 \end{array}$$

Como o dividendo continua menor que o divisor, colocamos outro 0 e "baixamos" mais um número.

$$\begin{array}{r} 111010 \overline{) 11} \\ 0101 \quad 1001 \\ - 11 \quad \quad \quad \\ \hline 100 \end{array}$$

Agora o dividendo é maior do que o divisor. Então colocamos 1 no quociente, subtraímos o divisor do dividendo e "baixamos" o próximo número.

$$\begin{array}{r} 111010 \overline{) 11} \\ 0101 \quad 10011 \\ - 11 \quad \quad \quad \\ \hline 100 \\ - 11 \quad \quad \quad \\ \hline 1 \end{array}$$

Da mesma forma, subtraímos o divisor do dividendo. Como não temos mais como "baixar" outro número, chegamos ao fim com a divisão tendo resto 1.

#### 4.1. DIVISÃO EM OCTAL

Vamos agora efetuar a divisão na base octal. Como já mencionamos, o raciocínio é o mesmo das bases binária e decimal, considerando que não vamos usar os algarismos 8 e 9.

Para isso, vamos efetuar  $(443 \div 17)_8$ .

Para facilitar essa operação é bom colocarmos uma tabuada em octal do divisor (que é o 17), como a seguir:

17x1 =	17
17x2 =	36
17x3 =	55
17x4 =	74
17x5 =	113
17x6 =	132
17x7 =	151
17x10 =	170

$$\begin{array}{r} 443 \overline{) 17} \\ - 36 \quad 2 \\ \hline 6 \end{array}$$

O dividendo é maior do que o divisor. Então, pela tabuada, vemos a linha a ser usada é a segunda.

$$\begin{array}{r} 443 \overline{) 17} \\ - 36 \quad 23 \\ \hline 63 \end{array}$$

Agora baixamos o 3 e, com ajuda da tabuada, vemos que a linha a ser usada é a terceira.

$$\begin{array}{r} 443 \overline{) 17} \\ - 36 \quad 23 \\ \hline 63 \\ - 55 \\ \hline 6 \end{array}$$

Como não temos mais como dividir, o quociente é 23 e o resto da divisão é 6.



## 4.2. DIVISÃO EM HEXADECIMAL

Nesta última operação aritmética vamos repetir o processo usado na base octal e criar uma tabuada do divisor para facilitar.

Como exemplo, vamos efetuar  $(B54A \div C1)_{16}$ .

Para facilitar essa operação é bom colocarmos uma tabuada em octal do divisor (que é o C1), como a seguir:

$C1 \times 1 = C1$   
 $C1 \times 2 = 182$   
 $C1 \times 3 = 243$   
 $C1 \times 4 = 304$   
 $C1 \times 5 = 3C5$   
 $C1 \times 6 = 486$   
 $C1 \times 7 = 547$   
 $C1 \times 8 = 608$   
 $C1 \times 9 = 6C9$   
 $C1 \times A = 78A$   
 $C1 \times B = 84B$   
 $C1 \times C = 90C$   
 $C1 \times D = 9CD$   
 $C1 \times E = A8E$   
 $C1 \times F = B4F$   
 $C1 \times 10 = C10$

$$\begin{array}{r} B54A \overline{) C1} \\ - B4F \quad F \\ \hline 5 \end{array}$$

O dividendo é maior do que o divisor. Então, pela tabuada, vemos a linha a ser usada é a décima quinta.

$$\begin{array}{r} B54A \overline{) C1} \\ - B4F \quad F0 \\ \hline 5A \end{array}$$

Agora baixamos o A e, como o novo dividendo é menor do que o divisor, colocamos 0 no quociente e o resto da divisão fica sendo 5A.

## CONSIDERAÇÕES FINAIS

Nesta aula vimos como trabalhar as quatro operações aritméticas nas bases binária, octal e hexadecimal.

São algoritmos semelhantes usados nas três bases (assim como na base decimal) e, embora pareça complicado a princípio, vemos com a prática que é muito simples trabalhar essas operações.

O mais importante é que, como qualquer atividade ligada à matemática, é fundamental a prática e repetição de exercícios para consolidar o aprendizado.

Como sugestão, utilize a calculadora do Windows, na versão programador, e pratique bastante até consolidar seu conhecimento.

Na próxima aula veremos estudaremos as portas lógicas que compõem os circuitos existentes no computador.

Até lá!

## MATERIAIS COMPLEMENTARES

Assista a esse vídeo bem didático com exemplos de multiplicação em octal. Link: <https://www.youtube.com/watch?v=3kStGiYGmkc&t=44s>.

Nesse material você poderá ler um pouco sobre a importância da base hexadecimal. Link: <https://canaltech.com.br/produtos/O-que-e-sistema-hexadecimal/>.

## REFERÊNCIAS

STALLINGS, William. *Arquitetura e organização de computadores: projeto para o desempenho*. 8ª edição. Editora Pearson. Livro (642 p.). ISBN 9788576055648. Disponível em: <<https://middleware-bv.am4.com.br/SSO/iesb/9788576055648>>. Acesso em: 16 out. 2022.

TANENBAUM, Andrew S. *Sistemas operacionais modernos*. 3ª edição. Editora Pearson. Livro (674 p.). ISBN 9788576052371. Disponível em: <<https://middleware-bv.am4.com.br/SSO/iesb/9788576052371>>. Acesso em: 16 out. 2022.

TANENBAUM, Andrew S. *Organização estruturada de computadores*. 6ª edição. Editora Pearson. Livro (628 p.). ISBN 9788581435398. Disponível em: <<https://middleware-bv.am4.com.br/SSO/iesb/9788581435398>>. Acesso em: 16 out. 2022.