

# Vamos pensar um pouco

A tabela ASCII - é o padrão utilizado para as informações

American Standard Code for Information Interchange

Binário	D	H	G	Binário	D	H	G	Binário	D	H	G
0010 0000	32	20	vazio	0100 0000	64	40	@	0110 0000	96	60	`
0010 0001	33	21	!	0100 0001	65	41	A	0110 0001	97	61	a
0010 0010	34	22	"	0100 0010	66	42	B	0110 0010	98	62	b
0010 0011	35	23	#	0100 0011	67	43	C	0110 0011	99	63	c
0010 0100	36	24	\$	0100 0100	68	44	D	0110 0100	100	64	d
0010 0101	37	25	%	0100 0101	69	45	E	0110 0101	101	65	e
0010 0110	38	26	&	0100 0110	70	46	F	0110 0110	102	66	f
0010 0111	39	27	'	0100 0111	71	47	G	0110 0111	103	67	g
0010 1000	40	28	(	0100 1000	72	48	H	0110 1000	104	68	h
0010 1001	41	29	)	0100 1001	73	49	I	0110 1001	105	69	i
0010 1010	42	2A	*	0100 1010	74	4A	J	0110 1010	106	6A	j
0010 1011	43	2B	+	0100 1011	75	4B	K	0110 1011	107	6B	k
0010 1100	44	2C	,	0100 1100	76	4C	L	0110 1100	108	6C	l
0010 1101	45	2D	-	0100 1101	77	4D	M	0110 1101	109	6D	m
0010 1110	46	2E	.	0100 1110	78	4E	N	0110 1110	110	6E	n
0010 1111	47	2F	/	0100 1111	79	4F	O	0110 1111	111	6F	o
0011 0000	48	30	0	0101 0000	80	50	P	0111 0000	112	70	p
0011 0001	49	31	1	0101 0001	81	51	Q	0111 0001	113	71	q
0011 0010	50	32	2	0101 0010	82	52	R	0111 0010	114	72	r
0011 0011	51	33	3	0101 0011	83	53	S	0111 0011	115	73	s
0011 0100	52	34	4	0101 0100	84	54	T	0111 0100	116	74	t
0011 0101	53	35	5	0101 0101	85	55	U	0111 0101	117	75	u
0011 0110	54	36	6	0101 0110	86	56	V	0111 0110	118	76	v
0011 0111	55	37	7	0101 0111	87	57	W	0111 0111	119	77	w
0011 1000	56	38	8	0101 1000	88	58	X	0111 1000	120	78	x
0011 1001	57	39	9	0101 1001	89	59	Y	0111 1001	121	79	y
0011 1010	58	3A	:	0101 1010	90	5A	Z	0111 1010	122	7A	z
0011 1011	59	3B	;	0101 1011	91	5B	[	0111 1011	123	7B	{
0011 1100	60	3C	<	0101 1100	92	5C	\	0111 1100	124	7C	
0011 1101	61	3D	=	0101 1101	93	5D	]	0111 1101	125	7D	}
0011 1110	62	3E	>	0101 1110	94	5E	^	0111 1110	126	7E	~
0011 1111	63	3F	?	0101 1111	95	5F	_				

128	Ç	144	É	160	á	176	░	192	┐	208	ƒ	224	ß	240	±
129	ü	145	æ	161	í	177	▒	193	└	209	ƒ	225	Γ	241	≥
130	é	146	Æ	162	ó	178	▓	194	┌	210	ƒ	226	π	242	≤
131	â	147	ô	163	û	179		195	─	211	ƒ	227	Σ	243	┌
132	ä	148	ö	164	ñ	180	└	196	┐	212	ƒ	228	σ	244	┐
133	à	149	ò	165	Ñ	181	└	197	┐	213	ƒ	229	μ	245	+
134	â	150	û	166	•	182	└	198	┐	214	ƒ	230	τ	246	≈
135	ç	151	ù	167	°	183	└	199	┐	215	ƒ	231	Φ	247	°
136	ê	152	—	168	¿	184	└	200	┐	216	ƒ	232	⊙	248	÷
137	ë	153	Ö	169	—	185	└	201	┐	217	ƒ	233	Ω	249	√
138	è	154	Û	170	┐	186	└	202	┐	218	ƒ	234	∞	250	—
139	í	156	£	171	½	187	└	203	┐	219	ƒ	235	φ	251	±
140	î	157	¥	172	¼	188	└	204	┐	220	ƒ	236	∞	252	—
141	ï	158	—	173	¡	189	└	205	┐	221	ƒ	237	ε	253	±
142	Ä	159	ƒ	174	«	190	└	206	┐	222	ƒ	238	ε	254	■
143	Å	192	Ł	175	»	191	└	207	┐	223	ƒ	239	∩	255	
								208	└	224	α	240	≡		

## Faça o seu primeiro nome em ASCII

### Observe:

Binary	Oct	Dec	Hex	Glyph		
				'63	'65	'67
010 0000	040	32	20	space		
010 0001	041	33	21	!		
010 0010	042	34	22	"		
010 0011	043	35	23	#		
010 0100	044	36	24	\$		
010 0101	045	37	25	%		
010 0110	046	38	26	&		
010 0111	047	39	27	'		
010 1000	050	40	28	(		
010 1001	051	41	29	)		
010 1010	052	42	2A	*		
010 1011	053	43	2B	+		
010 1100	054	44	2C	,		
010 1101	055	45	2D	-		
010 1110	056	46	2E	.		
010 1111	057	47	2F	/		
011 0000	060	48	30	0		
011 0001	061	49	31	1		
011 0010	062	50	32	2		
011 0011	063	51	33	3		
011 0100	064	52	34	4		
011 0101	065	53	35	5		
011 0110	066	54	36	6		

011 0111	067	55	37	7
011 1000	070	56	38	8
011 1001	071	57	39	9
011 1010	072	58	3A	:
011 1011	073	59	3B	;
011 1100	074	60	3C	<
011 1101	075	61	3D	=
011 1110	076	62	3E	>
011 1111	077	63	3F	?
100 0000	100	64	40	@
100 0001	101	65	41	A
100 0010	102	66	42	B
100 0011	103	67	43	C
100 0100	104	68	44	D
100 0101	105	69	45	E
100 0110	106	70	46	F
100 0111	107	71	47	G
100 1000	110	72	48	H
100 1001	111	73	49	I
100 1010	112	74	4A	J
100 1011	113	75	4B	K
100 1100	114	76	4C	L
100 1101	115	77	4D	M
100 1110	116	78	4E	N
100 1111	117	79	4F	O

100 1111	117	79	4F	O
101 0000	120	80	50	P
101 0001	121	81	51	Q
101 0010	122	82	52	R
101 0011	123	83	53	S
101 0100	124	84	54	T
101 0101	125	85	55	U
101 0110	126	86	56	V
101 0111	127	87	57	W
101 1000	130	88	58	X
101 1001	131	89	59	Y
101 1010	132	90	5A	Z
101 1011	133	91	5B	[
101 1100	134	92	5C	\
101 1101	135	93	5D	]
101 1110	136	94	5E	^
101 1111	137	95	5F	_
110 0000	140	96	60	@
110 0001	141	97	61	a
110 0010	142	98	62	b
110 0011	143	99	63	c
110 0100	144	100	64	d
110 0101	145	101	65	e
110 0110	146	102	66	f
110 0111	147	103	67	g
110 1000	150	104	68	h

110 1001	151	105	69	i
110 1010	152	106	6A	j
110 1011	153	107	6B	k
110 1100	154	108	6C	l
110 1101	155	109	6D	m
110 1110	156	110	6E	n
110 1111	157	111	6F	o
111 0000	160	112	70	p
111 0001	161	113	71	q
111 0010	162	114	72	r
111 0011	163	115	73	s
111 0100	164	116	74	t
111 0101	165	117	75	u
111 0110	166	118	76	v
111 0111	167	119	77	w
111 1000	170	120	78	x
111 1001	171	121	79	y
111 1010	172	122	7A	z
111 1011	173	123	7B	{
111 1100	174	124	7C	ACK
111 1101	175	125	7D	}
111 1110	176	126	7E	ESC

# Lógica Digital

## Portas lógicas (Gates)

As operações de um computador resumem-se na combinação de operações aritméticas básicas: somar, complementar, comparar e mover bits.

“Quem” realiza estas complicadíssimas operações são circuitos eletrônicos conhecidos como circuitos lógicos ou *Gates*. Os sistemas lógicos estão calcados na álgebra dos chaveamentos ou álgebra de Boole, instituída pelo matemático inglês George Boole (1815 – 1864) e que admite apenas duas grandezas: falso ou verdadeiro, representados por 0 e 1 respectivamente.

Esses sinais binários são representados por níveis de tensão nos circuitos do computador.

Os operadores lógicos ou funções lógicas básicas são as seguintes:

**E ou AND** – uma função é verdadeira se, e somente se, todos os termos forem verdadeiros

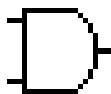
**Tabela Verdade**

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

### Expressão Booleana

$$X = A \cdot B$$

### Representação



**OU ou OR** – uma função é verdadeira se, qualquer um dos termos for verdadeiro

**Tabela Verdade**

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

**Expressão Booleana**

$$X = A + B$$

**Representação**



**NÃO ou NOT** – o termo é invertido

**Tabela Verdade**

A	X
0	1
1	0

**Expressão Booleana**

$$X = \bar{A}$$

**Representação**



**NÃO E ou NAND** – equivale a uma porta AND seguida de uma porta NÃO. O resultado é o inverso da saída de uma porta AND.

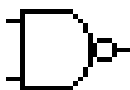
#### Tabela Verdade

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

#### Expressão Booleana

$$X = \overline{A \cdot B}$$

#### Representação



**NOR ou AND** – equivale a uma porta OR seguida de uma porta NÃO. O resultado é o inverso da saída de uma porta OR.

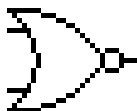
#### Tabela Verdade

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

#### Expressão Booleana

$$X = \overline{A + B}$$

#### Representação



**XOR ou OU EXCLUSIVO** – a função é verdadeira se, e somente se, um dos termos for verdadeiro.

**Tabela Verdade**

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

**Expressão Boleana**

$$X = A \oplus B$$

**Representação**

