APOSTILA
do Prof. Eduardo

Sistemas Digitais e Microprocessadores

Bases numéricas e códigos

Prof. Eduardo Furlan
2023

História da eletrônica digital

Primeiros sistemas

Máquina de Anticítera

Computador analógico e planetário mais antigo que se conhece

Século I a.C. na Grécia romana

Usado para prever posições astronômicas e eclipses, como função de calendário e astrologia

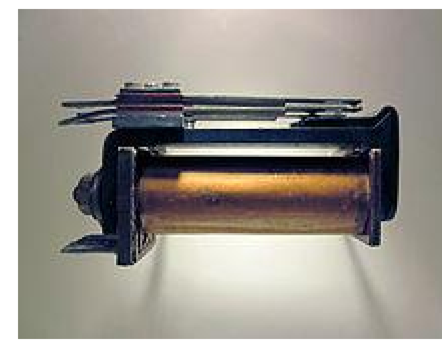
https://pt.wikipedia.org/wiki/M%C3%A1quina_de_Antic%C3%ADtera

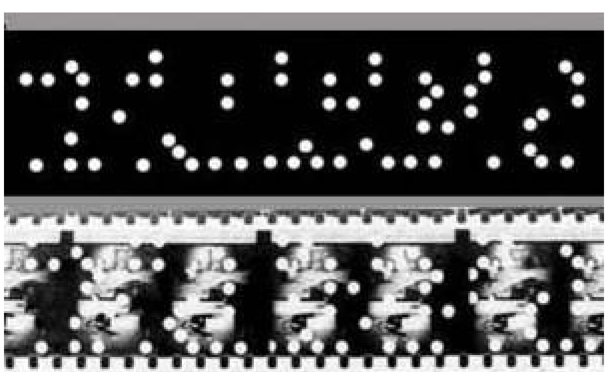
Z4 - computador eletromecânico

1º computador digital comercial

1942, Alemanha, por Konrad Zuse

2.500 relês

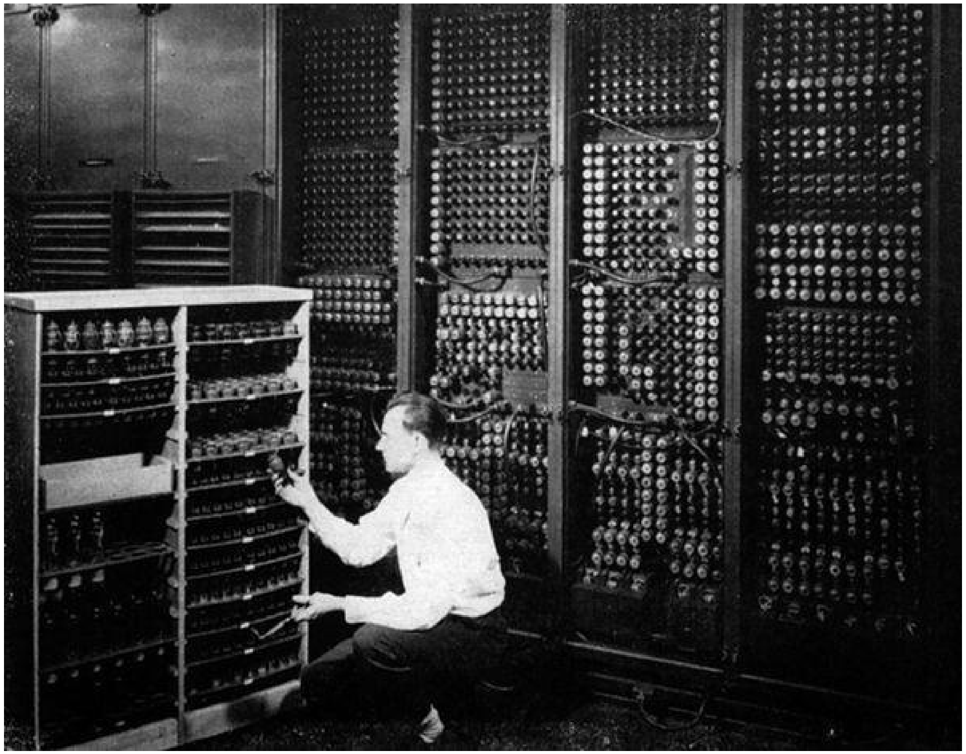




Programável perfurando bobinas descartadas de filme comum de 35mm

https://en.wikipedia.org/wiki/Z4_(computer)
https://machinemachine.net/text/ideas/on-text-and-exaptation/

ENIAC - computador eletrônico



1º eletrônico de grande porte

1945, EUA, por John Mauchly

18.000 válvulas termiônicas

https://www.computerhistory.org/revolution/birth-of-the-computer/4/78/325
https://pt.wikipedia.org/wiki/V%C3%A1lvula_termi%C3%B4nica

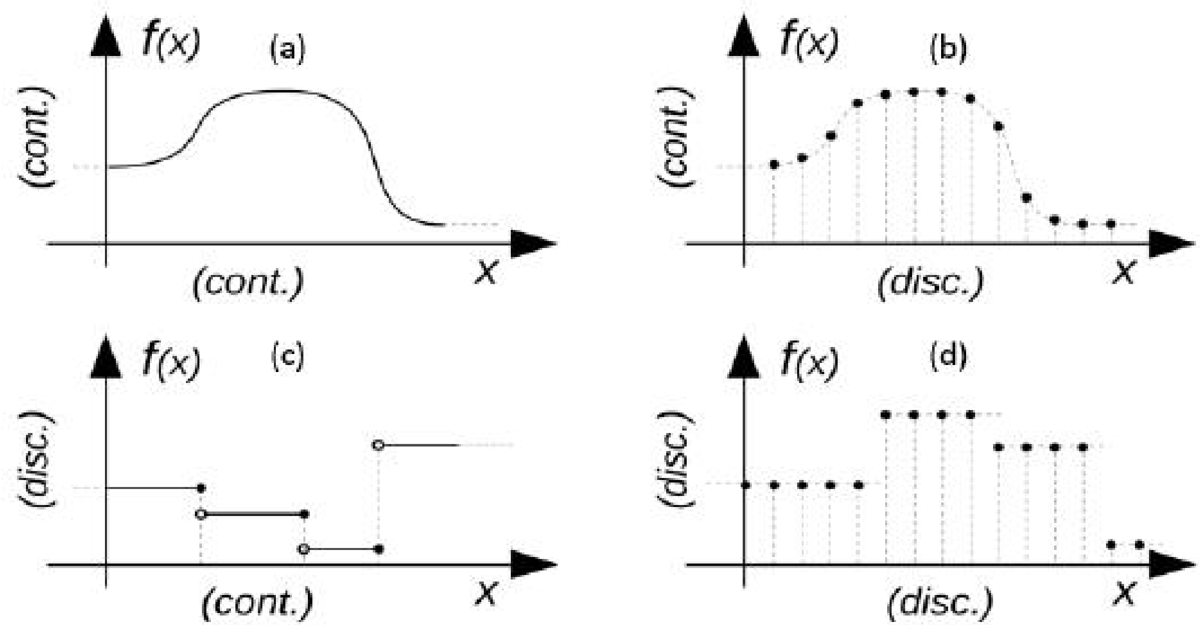
Frontier - supercomputer

2022, EUA, pela HPE

https://wikipedia.org/wiki/Frontier_(supercomputer)

https://pcworld.com

Grandezas analógicas e digitais



Grandezas contínuas e discretas

Discreto equivalente a digital, usando 2 estados, 0 e 1

Sistemas de numeração

Sistemas de escrita para expressar números

Exemplo

- 13 unidades de alguma coisa podem ser representadas
  - **13_{10}** na base decimal (10)
  - **d_{16}** na base hexadecimal (16)
  - **15_{8}** na base octal (8)
  - **1101_{2}** na base binária (2)

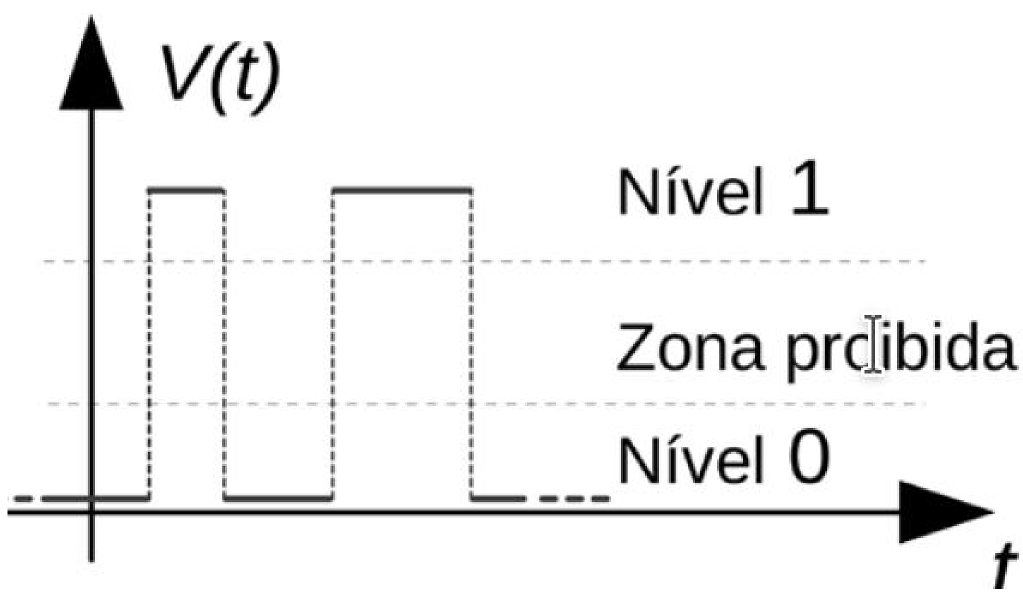
Sistema binário

Computadores internamente usam eletricidade em 2 estados que podem ser representados por

Ligado ou desligado, nível alto ou baixo, etc.

É conveniente o uso do sistema de numeração binário para representar

2 dígitos binários (bit) : 0 e 1



Base decimal

É a base que mais usamos no dia a dia

Usa 10 dígitos

- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Depois de 9, o próximo número é 10 (uma combinação)

Usamos outras bases no dia a dia:

- **60** - minutos, segundos, ângulos (360 = 6 x 60; 60 minutos de arco em um grau), e GPS (coordenadas geográficas)
  - Os dígitos são escritos usando a base decimal
- **12** ou **24** - horas

Números Romanos

É uma das formas de expressar números

I (1), V (5), X (10), L (50), C (100), D (500), M (1000)

Número	Número Romano	Cálculo
0	(não existe)	
1	I	1
2	II	1+1
3	III	1+1+1
4	IV	5-1
5	V	5
6	VI	5+1
7	VII	5+1+1
8	VIII	5+1+1+1
9	IX	10-1
10	X	10
11	XI	10+1

Sistema numérico decimal

Notação posicional

1 0 índice

posição

$$47_{10}$$

base

dígitos

$$= 4 \cdot 10 + 7 \cdot 1$$

$$= 4 \cdot 10^{1} + 7 \cdot 10^{0}$$

$$= 40 + 7$$

$$= 47_{10}$$

Classificação e categoria

Sistemas numéricos são classificados quanto ao uso de notação posicional, e ainda categorizados por **base**

Base 2 (binária) : 0, 1

Base 8 (octal) : 0, 1, 2, 3, 4, 5, 6, 7

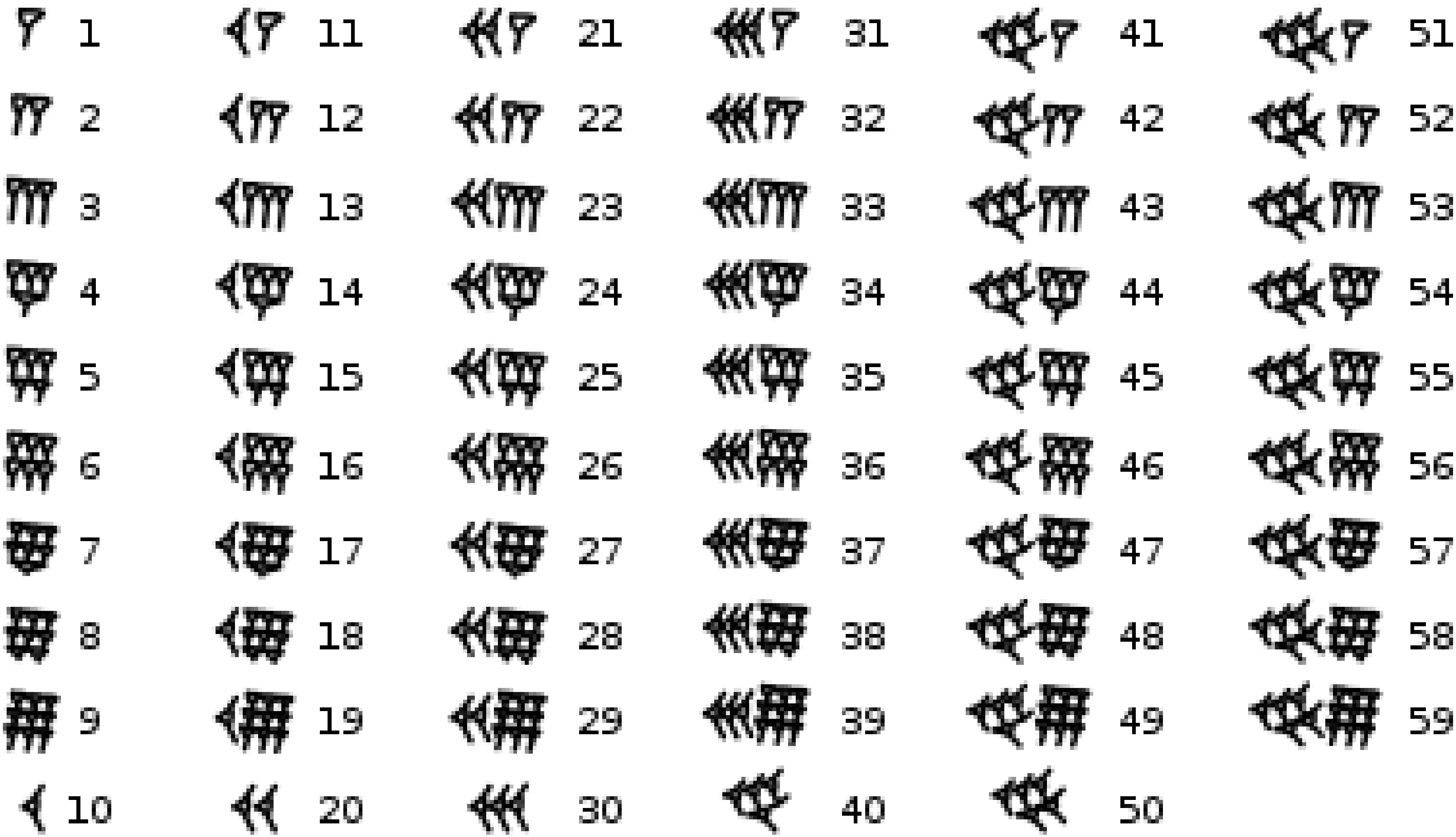
Base 10 (decimal) : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Base 16 (hexadecimal) : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f

etc.

Base 60 (sexagesimal)

Exemplo: numeração babilônica (escrita cuneiforme)



Base 10

Outro exemplo:

$$\overset{3}{2}\,\overset{2}{4}\,\overset{1}{5}\,\overset{0}{1}_{10}$$

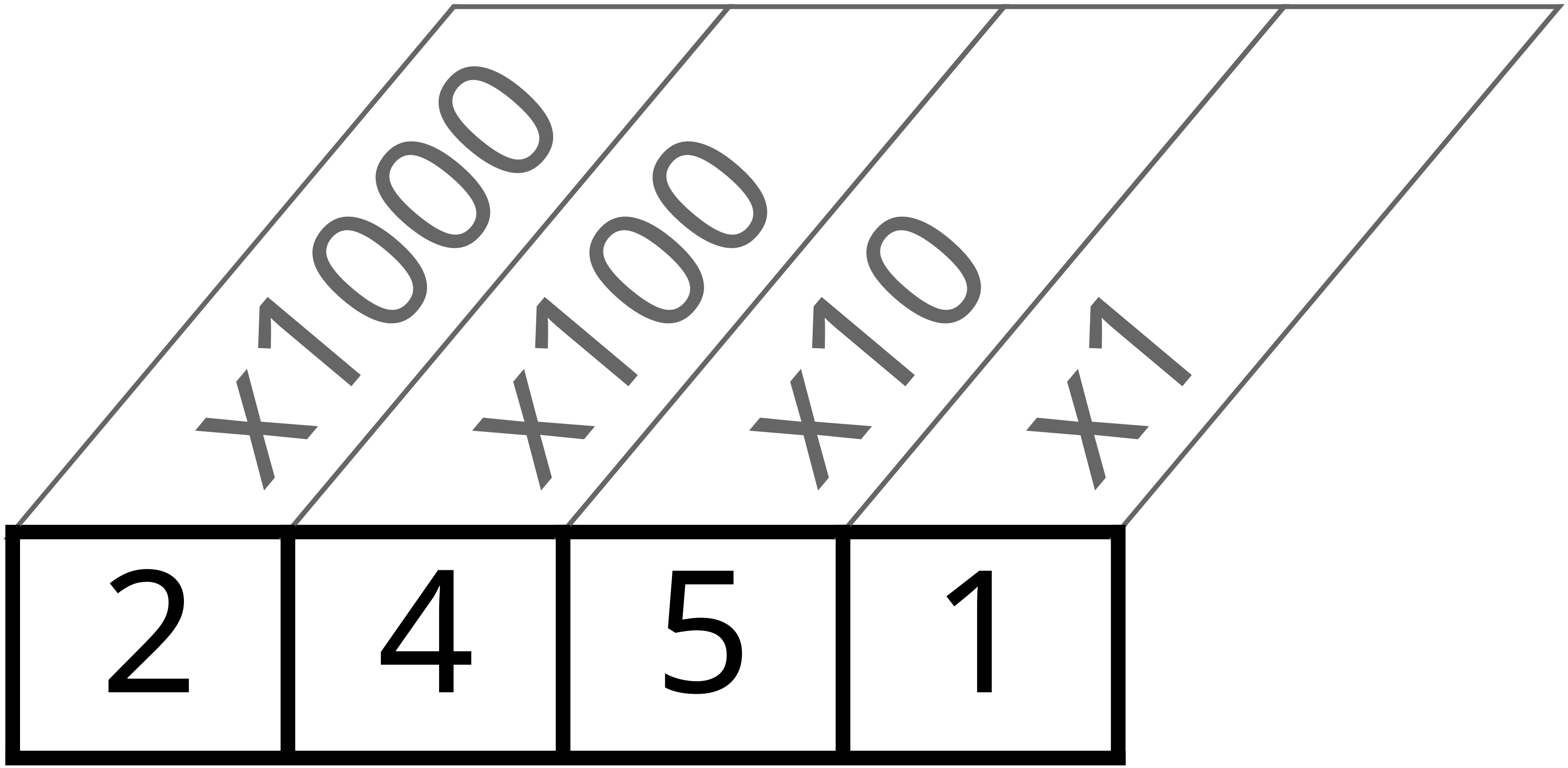
posição relativa, numericamente igual ao expoente, e no caso da base 10 também é equivalente à “quantidade de zeros”

$$= 2 \cdot 10^{3} + 4 \cdot 10^{2} + 5 \cdot 10^{1} + 1 \cdot 10^{0}$$

$$= 2\overset{3}{\boxed{000}} + 4\overset{2}{\boxed{00}} + 5\overset{1}{\boxed{0}} + 1\overset{0}{\boxed{}}$$

$$= 2451_{10}$$

Base 10



$= 2451_{10}$

Base 10

10 dígitos (0 a 9) :
0, 1, 2, 3, 4, 5, 6, 7, 8, 9

0
1
2
3
4
5
6
7
8
$\boxed{\begin{array}{c} 9 \\ 10 \end{array}}$
11
12
...

...
98
$\boxed{\begin{array}{c} 99 \\ 100 \end{array}}$
101
102
...

Base 8

8 dígitos (0 a 7) :
0, 1, 2, 3, 4, 5, 6, 7

0	13	...
1	14	76
2	15	77
3	16	100
4	17	101
5	20	102
6	21	...
7	22	
10	23	
11	24	
12	25	
	...	

Base 2

2 dígitos (0 a 1) :
0, 1

0
1
10
11
100
101
110
111
1000
1001
1010
1011
1100

Base 16

16 dígitos (0 a F) :
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f

0	8
1	9
2	a
3	b
4	c
5	d
6	e
7	f
	10
	11
	12
	...

```
[8]: for i in range(20) :
         print(f"{i:2}  {hex(i):4}  {oct(i):4}  {bin(i):4}")
```

```
 0   0x0    0o0    0b0
 1   0x1    0o1    0b1
 2   0x2    0o2    0b10
 3   0x3    0o3    0b11
 4   0x4    0o4    0b100
 5   0x5    0o5    0b101
 6   0x6    0o6    0b110
 7   0x7    0o7    0b111
 8   0x8    0o10   0b1000
 9   0x9    0o11   0b1001
10   0xa    0o12   0b1010
11   0xb    0o13   0b1011
12   0xc    0o14   0b1100
13   0xd    0o15   0b1101
14   0xe    0o16   0b1110
15   0xf    0o17   0b1111
16   0x10   0o20   0b10000
17   0x11   0o21   0b10001
18   0x12   0o22   0b10010
19   0x13   0o23   0b10011
```

$11_{10} = b_{16} = 13_8 = 1011_2$

0x, 0o, 0b, no caso da linguagem de programação, significa hexadecimal, octal, e binário

Algumas bases são convenientes para representar números quando se usa computadores

Ex.: um computador armazena 8 bits em um byte

O maior número que se pode armazenar em 8 bits é

$255_{10} = 11111111_2 = ff_{16}$

Neste caso a base hexadecimal usa apenas 2 dígitos

7	6	5	4	3	2	1	0	posição
1	1	1	1	1	1	1	1	binário
f				f				hexadecimal

Sistema hexadecimal

Exemplo

$$\overset{3}{2}\overset{2}{1}\overset{1}{a}\overset{0}{f}_{16}$$

$$= 2 \cdot 16^3 + 1 \cdot 16^2 + 10 \cdot 16^1 + 15 \cdot 16^0$$

$$= 8192 + 256 + 160 + 15$$

$$= 8623_{10}$$

Sistema octal

Exemplo

$$\overset{3}{4}\overset{2}{1}\overset{1}{7}\overset{0}{2}_{8}$$

$$= 4 \cdot 8^3 + 1 \cdot 8^2 + 7 \cdot 8^1 + 2 \cdot 8^0$$

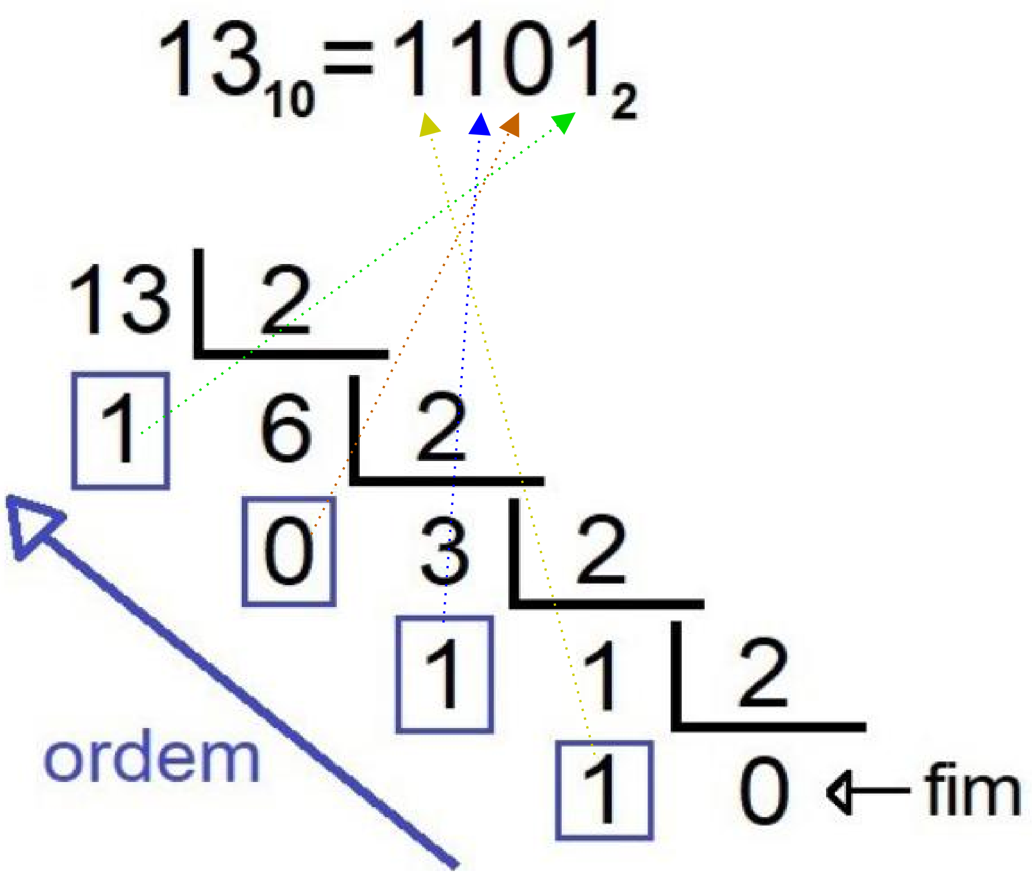
$$= 2048 + 64 + 56 + 2$$

$$= 2170_{10}$$

Conversão de bases

Método das divisões sucessivas

Exemplo: converter 13 decimal para binário



Parte fracionária

converser 0,3 em binário

$0{,}3 \cdot 2 = 0{,}6$

$0{,}6 \cdot 2 = 1{,}2$

$0{,}2 \cdot 2 = 0{,}4$

$0{,}4 \cdot 2 = 0{,}8$

$0{,}8 \cdot 2 = 1{,}6$

$0{,}6 \cdot 2 = 1{,}2$

$0{,}3_{10} = 0{,}010011_2$

Notação de excesso (ou bias)

Define-se o número de bits a ser utilizado e o valor de bias (excesso)

2^{n-1} valores negativos

$2^{n-1} - 1$ valores positivos

1 valor zero (1000)

0000	-8	1000	0
0001	-7	1001	1
0010	-6	1010	2
0011	-5	1011	3
0100	-4	1100	4
0101	-3	1101	5
0110	-2	1110	6
0111	-1	1111	7

n posições

7	6	5	4	3	2	1	0	posição
1	1	1	1	1	1	1	1	binário

bit de sinal

byte (8 bits)

Notação de complemento de dois

2^{n-1} valores negativos

$2^{n-1} - 1$ valores positivos

1 valor zero (0000)

0000	0	1000	-8
0001	1	1001	-7
0010	2	1010	-6
0011	3	1011	-5
0100	4	1100	-4
0101	5	1101	-3
0110	6	1110	-2
0111	7	1111	-1

n posições

7	6	5	4	3	2	1	0	posição
1	1	1	1	1	1	1	1	binário

bit de sinal

byte (8 bits)

Obtenção do valor negativo

Ex.: número 5

$5_{10} = 0101_2$

para obter -5 :

inverte 0101 obtendo 1010

soma 1 ao resultado

1010

\+   1

1011

Notação de complemento de dois

0000	0	1000	-8
0001	1	1001	-7
0010	2	1010	-6
0011	3	1011	-5
0100	4	1100	-4
0101	5	1101	-3
0110	6	1110	-2
0111	7	1111	-1

Código BCD

BCD - *Binary Coded Decimal*

Cada algarismo em decimal é convertido para binário

5	4	7	decimal
0101	0100	0111	BCD

Código de Gray

Código decimal	Código Binário	Código Gray
0	0	0
1	1	1
2	10	11
3	11	10
4	100	110
5	101	111
6	110	101
7	111	100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Surgiu na época das válvulas

Tentava resolver problemas com ruído quando vários bits eram modificados simultaneamente

Cada mudança sequencial muda apenas um bit

https://pt.wikipedia.org/wiki/C%C3%B3digo_de_Gray

EBCDIC

Padrão criado pela IBM nos anos 1960

Baseado no BCD

Primeira tentativa de normalização em paralelo com a normalização ASCII

https://pt.wikipedia.org/wiki/Extended_Binary_Coded_Decimal_Interchange_Code

Padrão ASCII

ASCII control characters

00	NULL	(Null character)
01	SOH	(Start of Header)
02	STX	(Start of Text)
03	ETX	(End of Text)
04	EOT	(End of Trans.)
05	ENQ	(Enquiry)
06	ACK	(Acknowledgement)
07	BEL	(Bell)
08	BS	(Backspace)
09	HT	(Horizontal Tab)
10	LF	(Line feed)
11	VT	(Vertical Tab)
12	FF	(Form feed)
13	CR	(Carriage return)
14	SO	(Shift Out)
15	SI	(Shift In)
16	DLE	(Data link escape)
17	DC1	(Device control 1)
18	DC2	(Device control 2)
19	DC3	(Device control 3)
20	DC4	(Device control 4)
21	NAK	(Negative acknowl.)
22	SYN	(Synchronous idle)
23	ETB	(End of trans. block)
24	CAN	(Cancel)
25	EM	(End of medium)
26	SUB	(Substitute)
27	ESC	(Escape)
28	FS	(File separator)
29	GS	(Group separator)
30	RS	(Record separator)
31	US	(Unit separator)
127	DEL	(Delete)

ASCII printable characters

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

Extended ASCII characters

128	Ç	160	á	192	└	224	Ó
129	ü	161	í	193	┴	225	ß
130	é	162	ó	194	┬	226	Ô
131	â	163	ú	195	├	227	Ò
132	ä	164	ñ	196	─	228	õ
133	à	165	Ñ	197	┼	229	Õ
134	å	166	ª	198	ã	230	µ
135	ç	167	º	199	Ã	231	þ
136	ê	168	¿	200	╚	232	Þ
137	ë	169	®	201	╔	233	Ú
138	è	170	¬	202	╩	234	Û
139	ï	171	½	203	╦	235	Ù
140	î	172	¼	204	╠	236	ý
141	ì	173	¡	205	═	237	Ý
142	Ä	174	«	206	╬	238	¯
143	Å	175	»	207	¤	239	´
144	É	176	░	208	ð	240	≡
145	æ	177	▒	209	Ð	241	±
146	Æ	178	▓	210	Ê	242	‗
147	ô	179	│	211	Ë	243	¾
148	ö	180	┤	212	È	244	¶
149	ò	181	Á	213	ı	245	§
150	û	182	Â	214	Í	246	÷
151	ù	183	À	215	Î	247	¸
152	ÿ	184	©	216	Ï	248	°
153	Ö	185	╣	217	┘	249	¨
154	Ü	186	║	218	┌	250	·
155	ø	187	╗	219	█	251	¹
156	£	188	╝	220	▄	252	³
157	Ø	189	¢	221	¦	253	²
158	×	190	¥	222	Ì	254	■
159	ƒ	191	┐	223	▀	255	nbsp

Código Excess-3

Decimal	Excess-3
−3	0000
−2	0001
−1	0010
0	**0011**
1	**0100**
2	**0101**
3	**0110**
4	**0111**
5	**1000**
6	**1001**
7	**1010**
8	**1011**
9	**1100**
10	1101
11	1110
12	1111

Usado em computadores antigos, máquinas registradoras, calculadoras da década de 70, etc.

127_{10} = 0100 0101 1010

Permite simplificar o hardware para fazer cálculos

Não usa 0000 e 1111 para representar um dígito (uma falha de hardware pode eventualmente gerar estes números)

https://en.wikipedia.org/wiki/Excess-3

APOSTILA
do Prof. Eduardo

https://github.com/efurlanm/teaching/

Prof. Eduardo Furlan
2023