

# Modelo de Envio de Exercícios

Email: [marcioklein@uni9.pro.br](mailto:marcioklein@uni9.pro.br)

Subject (Assunto): 25/08/2022 – 5ª f DIURNO -

RA<sub>1</sub> - Nome<sub>1</sub>

RA<sub>2</sub> - Nome<sub>2</sub>

RA<sub>3</sub> - Nome<sub>3</sub>

RA<sub>4</sub> - Nome<sub>4</sub>

◦

◦

◦

RESOLUÇÃO DO EXERCÍCIO

# Tabela ASCII

**ASCII** (*American Standard Code for Information Interchange*; "Código Padrão Americano para o Intercâmbio de Informação") — é um código binário (cadeias de bits: 0s e 1s) que codifica um conjunto de 128 sinais: 95 sinais gráficos (letras do alfabeto latino, sinais de pontuação e sinais matemáticos) e 33 sinais de controle, utilizando portanto apenas 7 bits para representar todos os seus símbolos. Observe a tabela a seguir.

O 8º bit pode ser usado como check-digit.

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

# Tabela ASCII

- Observe as três colunas que compõe a tabela:
  - A primeira (Dec) é a velha conhecida nossa.
  - A segunda coluna (Hex), com valores decimais e letra, será objeto do nosso próximo estudo.
  - A terceira coluna é dos caracteres que correspondem a cada um dos código binário da tabela (**binário? Como assim** )

# Tabela Hexadecimal

Binário	Hex
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binário	Hex
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

# Conversão de caractere em byte

De acordo com  
a Tabela ASCII  
temos que:  
Exemplos

Aplicando  
a Tabela ASCII  
para binário  
temos que:

Caractere	Hex		Byte
*	2A	→	00101010
>	3E	→	00111110
M	4D	→	01001101
k	6B	→	01101011

# Usos da base hexadecimal

- Desta forma, quando o computador quiser enviar a letra M (eme maiúsculo) para o vídeo ou para a impressora, ele enviará o byte composto pelos bits 01001101
- Mas não é só para representar caracteres que a base hexadecimal é usada. Posto que as instruções do microprocessador são também binárias, a tabela de instruções de um microprocessador também utiliza a representação hexadecimal, conforme pode-se observar no próximo slide (tabela de instruções do microprocessador Intel 8080)



## Subconjunto de Instruções do Processador 8080

Hex	Mnemonic	Hex	Mnemonic	Hex	Mnemonic
00	NOP	2B	DCX H	56	MOV D,M
01	LXI B,D16	2C	INR L	57	MOV D,A
02	STAX B	2D	DCR L	58	MOV E,B
03	INX B	2E	MVI L,D8	59	MOV E,C
04	INR B	2F	CMA	5A	MOV E,D
05	DCR B	30	. . .	5B	MOV E,E
06	MVI B,D8	31	LXI SP,D16	5C	MOV E,H
07	RLC	32	STA Adr	5D	MOV E,L
08	. . .	33	INX SP	5E	MOV E,M
09	DAD B	34	INR M	5F	MOV E,A
0A	LDAX B	35	DCR M	60	MOV H,B
0B	DCX B	36	MVI M,D8	61	MOV H,C



# Usos da base hexadecimal

- A base hexadecimal também é usada para representar os endereços (posições) de memória, bem como os respectivos conteúdos destes, como mostrado no próximo slide.

# Programa Java e Arquivo de Classe Compila- do

0000	cafe	babe	0003	002d	0012	0700	0e07	0010	.....
0010	0a00	0200	040c	0007	0005	0100	0328	2956	.....()
0020	0100	1628	5b4c	6a61	7661	2f6c	616e	672f	...([Ljava/lang/
0030	5374	7269	6e67	3b29	5601	0006	3c69	6e69	String;)V...<ini
0040	743e	0100	0443	6f64	6501	000d	436f	6e73	t>...Code...Cons
0050	7461	6e74	5661	6c75	6501	000a	4578	6365	tantValue...Exce
0060	7074	696f	6e73	0100	0f4c	696e	654e	756d	ptions...LineNum
0070	6265	7254	6162	6c65	0100	0e4c	6f63	616c	berTable...Local
0080	5661	7269	6162	6c65	7301	000a	536f	7572	Variables...Sour
0090	6365	4669	6c65	0100	0361	6464	0100	0861	ceFile...add...a
00a0	6464	2e6a	6176	6101	0010	6a61	7661	2f6c	dd.java...java/l
00b0	616e	672f	4f62	6a65	6374	0100	046d	6169	ang/Object...mai
00c0	6e00	2100	0100	0200	0000	0000	0200	0900	n.....
00d0	1100	0600	0100	0800	0000	2d00	0200	0400	.....
00e0	0000	0d10	0f3c	1009	3d03	3e1b	1c60	3eb1	.....
00f0	0000	0001	000b	0000	000e	0003	0000	0004	.....
0100	0008	0006	000c	0002	0001	0007	0005	0001	.....
0110	0008	0000	001d	0001	0001	0000	0005	2ab7	.....

## DECIFRANDO A TABELA

	00001	0203	0405	0607	0809	A0B0	C0D0	E0F0	ASCII V	ASCII i
0000	cafe	babe	0003	002d	0012	0700	0e07	0010	.....	
0010	0a00	0200	040c	0007	0005	0100	0328	2956	.....()	V
0020	0100	1628	5b4c	6a61	7661	2f6c	616e	672f	...([Ljava/lang/	
0030	5374	7269	6e67	3b29	5601	0006	3c69	6e69	String;)V...<ini	
0040	743e	0100	0443	6f64	6501	000d	436f	6e73	t>...Code...Cons	
0050	7461	6e74	5661	6c75	6501	000a	4578	6365	tantValue...Exce	
0060	7074	696f	6e73	0100	0f4c	696e	654e	756d	ptions...LineNum	
0070	6265	7254	6162	6c65	0100	0e4c	6f63	616c	berTable...Local	
0080	5661	7269	6162	6c65	7301	000a	536f	7572	Variables...Sour	
0090	6365	4669	6c65	0100	0361	6464	0100	0861	ceFile...add...a	
00a0	6464	2e6a	6176	6101	0010	6a61	7661	2f6c	dd.java...java/l	
00b0	616e	672f	4f62	6a65	6374	0100	046d	6169	ang/Object...mai	
00c0	6e00	2100	0100	0200	0000	0000	0200	0900	n.....	
00d0	1100	0600	0100	0800	0000	2d00	0200	0400	.....	
00e0	0000	0d10	0f3c	1009	3d03	3e1b	1c60	3eb1	.....	
00f0	0000	0001	000b	0000	000e	0003	0000	0004	.....	
0100	0008	0006	000c	0002	0001	0007	0005	0001	.....	
0110	0008	0000	001d	0001	0001	0000	0005	2ab7	.....	
0120	0003	b100	0000	0100	0b00	0000	0600	0100	.....	
0130	0000	0100	0100	0d00	0000	0200	0f00		.....	

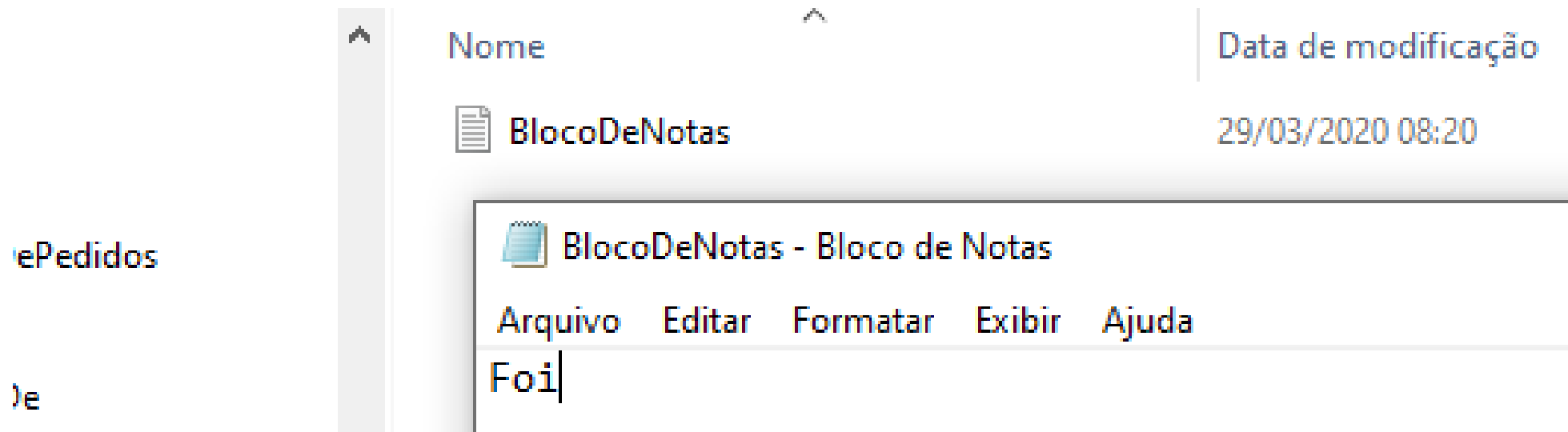
# Usos da base hexadecimal

- Finalmente, com código hexadecimal e um editor hexadecimal como o Hex Editor Neo, podemos conhecer e alterar o conteúdo de quaisquer arquivos do HD.
- Por exemplo, suponhamos a criação de um arquivo através do bloco de notas, chamado Foi.txt, cujo conteúdo é exatamente a palavra “Foi” conforme mostra o próximo slide.

# Usos da base hexadecimal

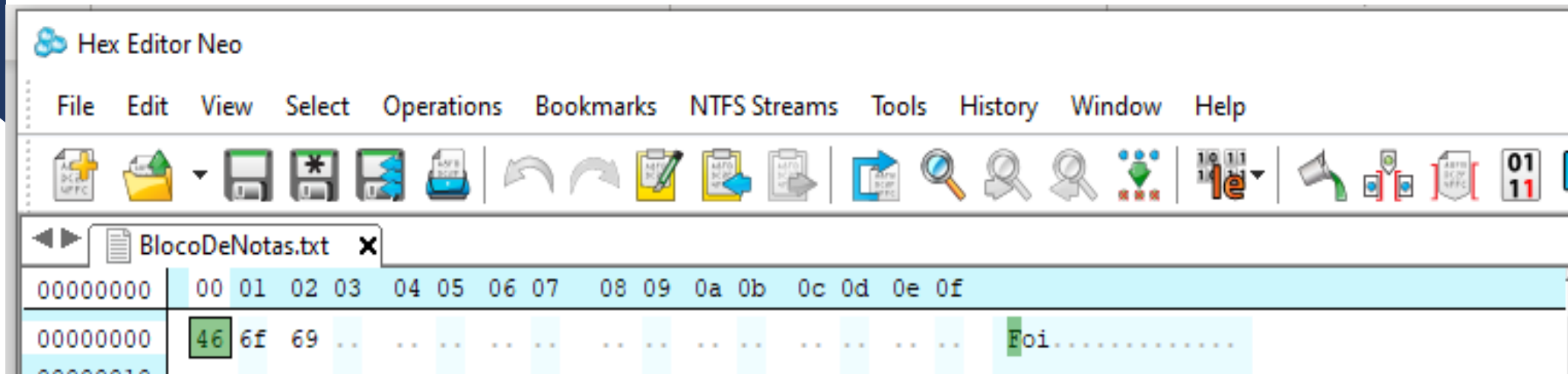
- O arquivo de nome BlocoDeNotas.txt é gravado no diretório
  - c:/Este Computador/Winows(c:)/TesteDoEditorHexa.

Este Computador > Windows (C:) > TesteDoEditorHexa



# Usos da base hexadecimal

- Se abrirmos o mesmo arquivo com o editor Hexa teremos:



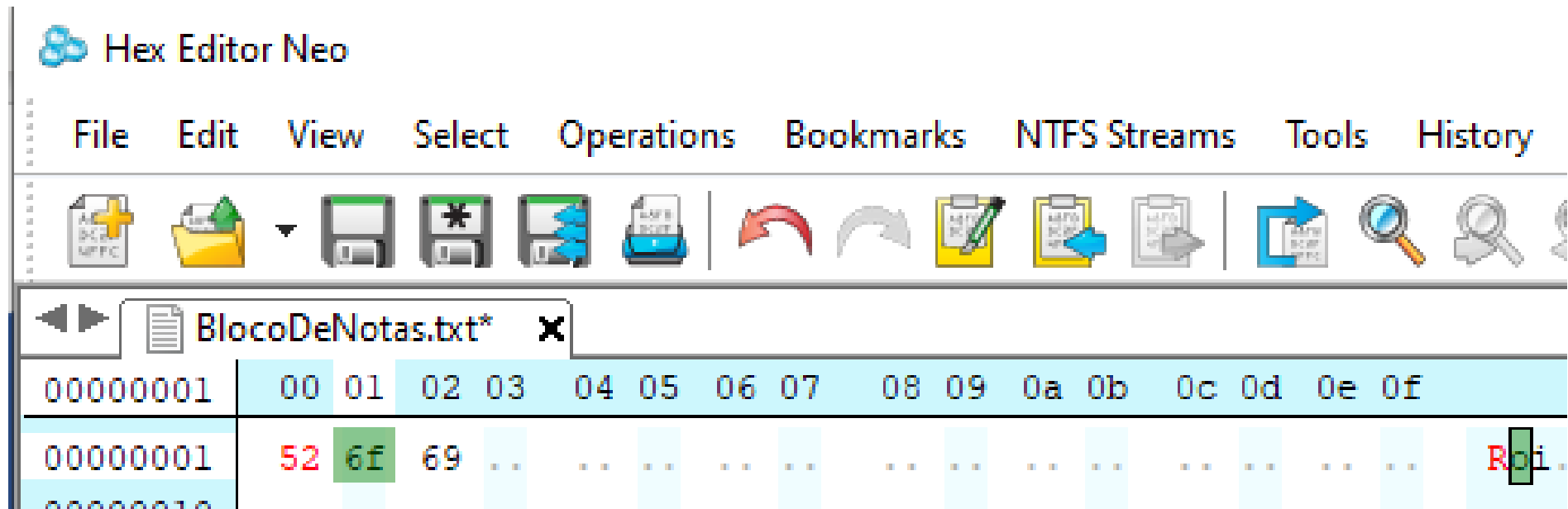
- Isso nos indica que:

END DISCO	ASCII	HEXA	BINÁRIO (memória do computador)
0000000000	F	46	0100 0110
0000000001	o	6f	0110 1111
0000000002	i	69	0110 1001



# Usos da base hexadecimal

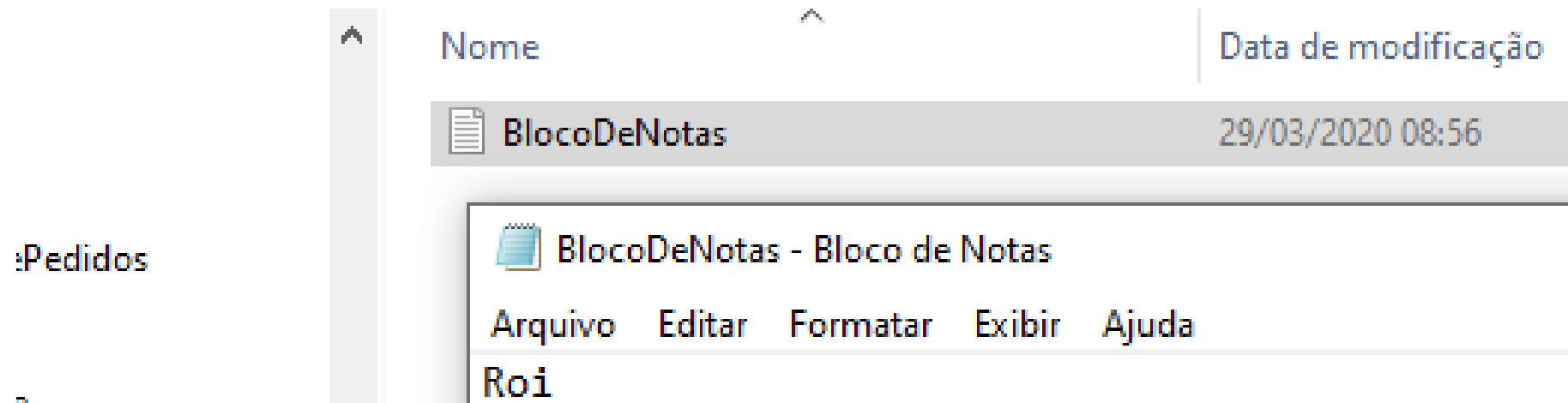
- Se mudarmos o F para R teremos



# Usos da base hexadecimal

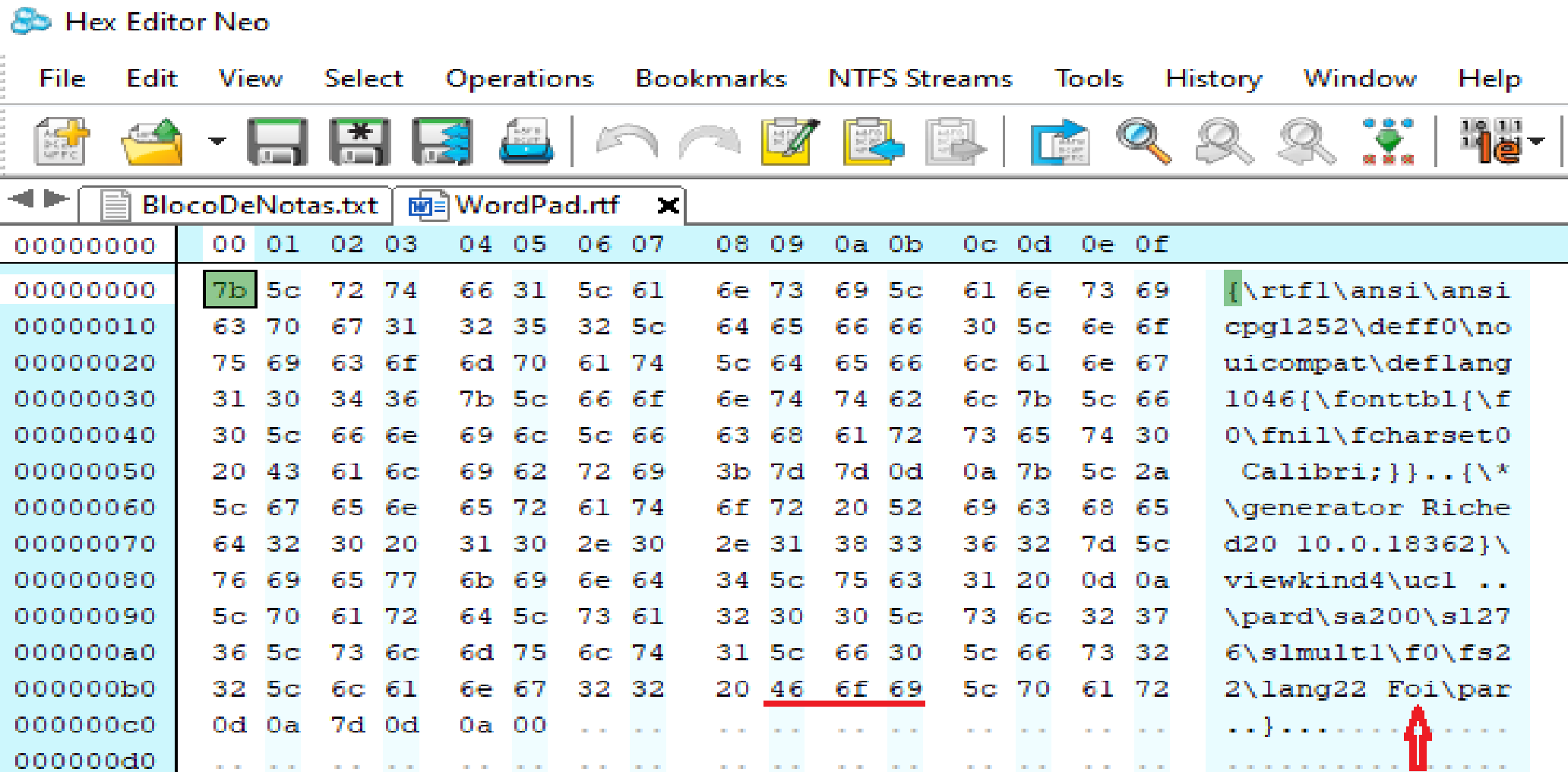
- Se gravarmos e voltarmos ao editor, teremos:

Este Computador > Windows (C:) > TesteDoEditorHexa



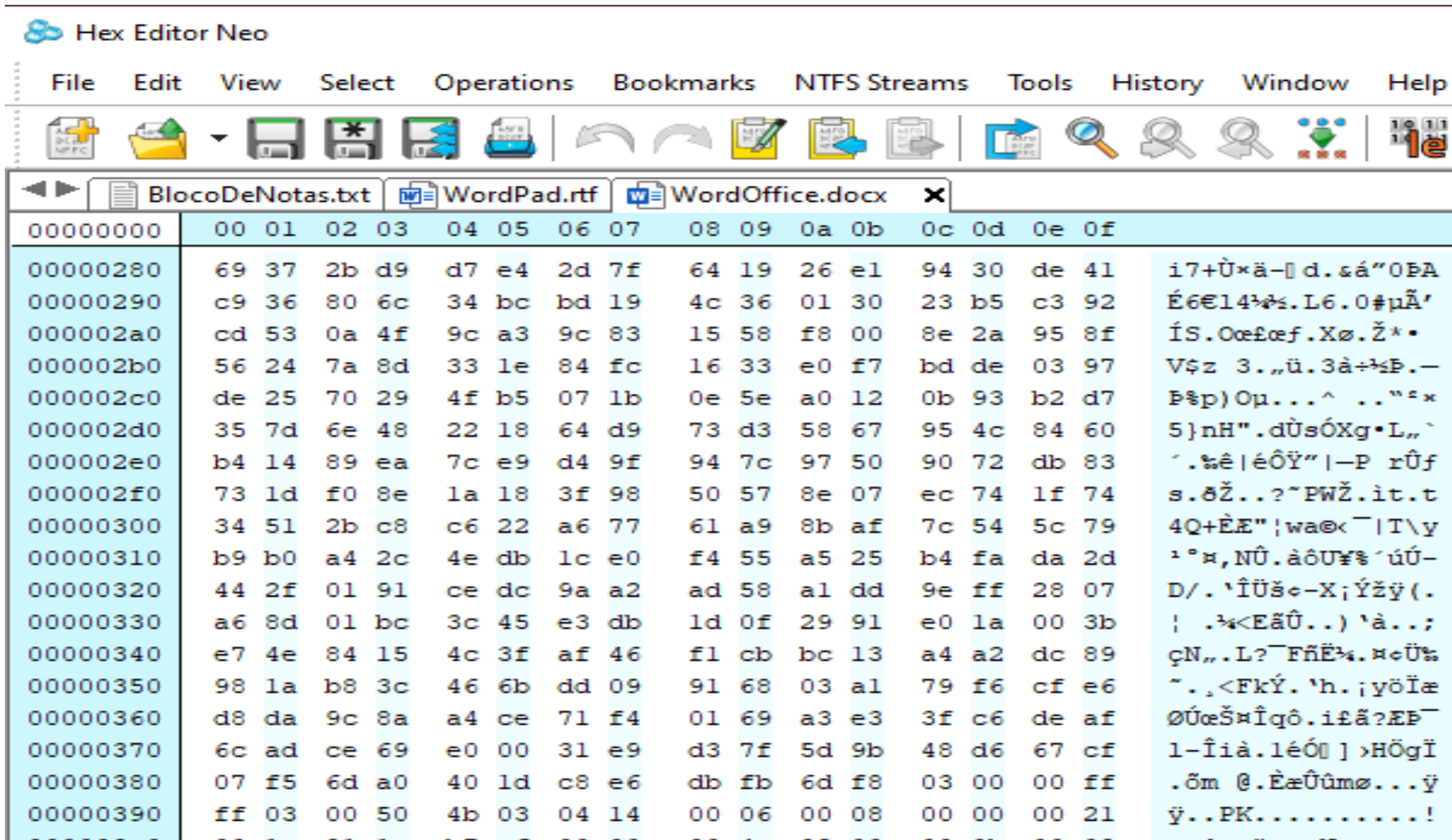
# Usos da base hexadecimal

- Criando um arquivo semelhante no WordPad e lendo com o hexa edito:



# Usos da base hexadecimal

- Por último, com o Word Office, que não segue o padrão ASCII:



# Voltando ao universo da Matemática

- Recordando a tabela hexadecimal:

Binário	Hex
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binário	Hex
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

# Conversão de hexadecimal em decimal

- Procure verificar os decimais correspondentes a A, B, C, D, E, F:

Binário	Hex	Dec
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Binário	Hex	Dec
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Assim, vamos ver, por exemplo, a conversão de  $3F_{16}$  em decimal:

$$\begin{array}{c|c} 3 & F \\ \hline 16^1 & 16^0 \end{array} \Rightarrow 3 \times 16^1 + F \times 16^0$$

$$\text{sendo } F_{16} \Rightarrow 15_{10} \Rightarrow$$

$$\Rightarrow 3 \times 16 + 15 \times 1 =$$

$$= 48 + 15 = 63_{10}$$



# Exercícios

7. Converter em decimais os seguintes hexadecimais:

a)  $1C3_{16} =$

b)  $238_{16} =$

c)  $1FC9_{16} =$

# Conversão Decimal para Hexadecimal

- Procure verificar os decimais correspondentes a A, B, C, D, E, F:

Binário	Hex	Dec
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Binário	Hex	Dec
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Diagram illustrating the conversion of the decimal number 1000 to hexadecimal using successive division by 16:

1000	16		
992	62	16	
8	48	3	16
	14	0	0
	E	3	

$1000_{10} = 3E8_{16}$

# Exercícios

8. Converter em hexadecimais os seguintes decimais:

a)  $134_{10} =$

b)  $384_{10} =$

c)  $3882_{10} =$

# Conversão Hexadecimal para Binário

- Procure verificar os decimais correspondentes a A, B, C, D, E, F:

Binário	Hex	Dec
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Binário	Hex	Dec
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Seja o valor Hexa **C13<sub>16</sub>**

Sua conversão para binário é bastante simples e muito semelhante à de Octal para binário, só que invés de cada valor corresponder à 3 bits, corresponde à 4 bits.

C      1      3  
1100 0001 0011

Ou seja:

**C13<sub>16</sub> = 110000010011<sub>2</sub>**

# Exercícios

9. Converter em hexadecimais para binários:

a)  $1ED_{16} =$

b)  $6CF9_{16} =$

c) Transformar o hexadecimal 3A7 em binário e o resultado em octal

# Conversão Binário para Hexadecimal

- Procure verificar os decimais correspondentes a A, B, C, D, E, F:

Binário	Hex	Dec
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7

Binário	Hex	Dec
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Análogo à conversão de Binário para Octal, porém com agrupamento de 4 dígitos.

**Exemplo:**

**10011000<sub>2</sub>**  $\Rightarrow$   
1001 1000  
9 8

$\Rightarrow$  **98<sub>16</sub>**



# Exercícios

10. Converter em hexadecimais os binários:

a)  $1100011_2 =$

b)  $11000111100011100_2 =$