

Exercice 1 : Compléter le tableau suivant:

décimal	binaire	hexadécimal	Base 4	base 8
53	110101	35	311	65
85	1010101	55	1111	125
		41E		
				12377
243				
	111010101			
		12CF		
			23031	

Exercice 2 : Effectuer directement les opérations suivantes (sans repasser par la base 10)

- $(1101)_2 + (111)_2$
- $(101101)_2 + (11100)_2$
- $(10011)_2 + (11010)_2$
- $(1011101)_2 + (100011)_2$
- $(111011)_2 - (10001)_2$
- $(10011)_2 - (1101)_2$
- $(10100)_2 - (1111)_2$
- $(1011)_2 - (101)_2$

Exercice 3 : Effectuer directement les opérations suivantes (sans repasser par la base 10)

- $(B7)_{16} + (35)_{16}$
- $(8A)_{16} + (C7)_{16}$
- $(2A5F)_{16} + (BE4)_{16}$
- $(B35D)_{16} + (3CA8)_{16}$
- $(B7)_{16} - (35)_{16}$
- $(C7)_{16} - (8A)_{16}$
- $(2A5F)_{16} - (BE4)_{16}$
- $(B35D)_{16} - (3CA8)_{16}$

Exercice 4 : Convertir en base 10 les nombres binaires suivants

- $(0,0111011)_2$
- $(110,101101)_2$
- $(1011,00101)_2$
- $(11,00100100001)_2$

Exercice 5 : Convertir en base 10 les nombres hexadécimaux suivants

- a. $(0,3B)_{16}$
- b. $(2A,C5)_{16}$
- c. $(7E0,A6D)_{16}$
- d. $(35,1C8F)_{16}$

Exercice 6 :

Convertir les nombres décimaux en base 2 (16 chiffres binaires après la virgule si cela ne tombe pas « juste ») et en base 16 (4 chiffres hexadécimaux après la virgule si cela ne tombe pas « juste »)

- a. 21,8359375
- b. 47,12
- c. 2,718
- d. 213,0146

Exercice 7 : Calculer les produits suivants en base 2

- a. $(11000)_2 * (11)_2$
- b. $(11011101)_2 * (11110011)_2$

Exercice 8 : Calculer les quotients suivants en base 2 (16 chiffres binaires après la virgule si cela ne tombe pas « juste »)

- a. $(11110100)_2 / (1101)_2$
- b. $(1011111)_2 / (100110)_2$

Exercice 9 : Donner les arrondis de

- a. $(1101011)_2$ à $(100)_2$ près
- b. $(10,011)_2$ à $(0,1)_2$ près
- c. $(A,BB)_{16}$ à $(0,1)_{16}$ près

Exercice Bonus : ASCII

Le code *ASCII American Standard Code for Information Interchange* permet de coder les caractères principaux utilisés en informatique. Le tableau suivant permet de retrouver le code *ASCII* des principaux caractères.

	<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>
<i>0</i>			SP	0	@	P		p
<i>1</i>			!	1	A	Q	a	q
<i>2</i>			«	2	B	R	b	r
<i>3</i>			#	3	C	S	c	s
<i>4</i>			\$	4	D	T	d	t
<i>5</i>			%	5	E	U	e	u
<i>6</i>			&	6	F	V	f	v
<i>7</i>			‘	7	G	W	g	w
<i>8</i>			(8	H	X	h	x
<i>9</i>)	9	I	Y	i	y
<i>A</i>			*	:	J	Z	j	z
<i>B</i>			+	;	K	[k	{
<i>C</i>			,	<	L	\	l	
<i>D</i>			-	=	M	_	m	}
<i>E</i>			.	>	N		n	
<i>F</i>			/	?	O	_	o	

Exemple de lecture :
Le code du caractère « A » est **\$41**

- a) Le code ASCII de ce tableau est-il donné en hexadécimale, décimale ou binaire.
- b) Donner le code du caractère « B » en binaire pur.
- c) Donner les 5 codes du mot « **ELORN** ».
- d) Quelle opération arithmétique doit-on réaliser pour convertir le code ASCII d'un caractère majuscule vers celui d'un minuscule. Donner des exemples.

L'adresse de cette donnée est : 1BD500C0

Adresse 1BD500EF

L'adresse de cette donnée est : 1BD500D1

Adresses	Données																
1BD500C0	50	20	20	20	20	20	20	20	20	20	20	20	30	3B	37	32	20
1BD500D0	20	20	20	20	20	35	36	20	20	20	20	20	20	20	20	20	28
1BD500E0	30	3B	31	34	31	29	20	20	20	20	C4	C4	2D	02	00	01	
1BD500F0	02	00	3C	50	47	2E	50	52	45	43	20	20	20	20	20	20	
1BD50100	36	C4	7F	04	26	8B	4D	08	26	29	4D	08	E3	19	26	C4	
1BD50110	7D	0C	80	3E	DE	00	00	75	0B	26	8A	05	E8	1C	20	22	
1BD50120	55	6E	20	76	65	72	72	65	20	64	65	20	42	69	64	75	
1BD50130	6C	65	22	20	47	E2	F7	EB	03	E8	9C	00	E8	DB	FC	33	
1BD50140	C0	CA	04	00	33	C0	CA	04	00	B0	0D	E8	02	00	B0	0A	
1BD50150	53	51	52	06	50	E8	72	00	58	3C	07	74	2A	3C	08	74	
1BD50160	2D	3C	0D	74	33	3C	0A	74	35	B4	09	8A	1E	E2	00	32	

- e) Retrouver le contenu de la variable de type chaîne de caractère sur 6 octets qui existe dans ce programme à partir de l'adresse suivante : 0001101111010101000000100101100 (adresse en binaire pur).