

Rédiger sur une copie en soignant la présentation (laisser de la place pour les observations)

Travail à faire par binôme (équipe de deux élèves)

1) La numération babylonienne

a) Le système de numération de la civilisation babylonienne¹ n'utilise que deux symboles : $\vee$ (valeur : 1) et $<$ (valeur : 10).

On peut juxtaposer jusqu'à 9 fois le symbole $\vee$ et 5 fois le symbole $<$.

Ainsi 12 s'écrit $<\vee\vee$ (10+2) et 36 s'écrit $<<<\vee\vee\vee\vee\vee$ (30+6).

☞ Comment écrit-on les nombres 2, 12, 24 et 59 en babylonien ?



b) Ce système de numération est un *système de positions* : la valeur des symboles est multipliée par 60 en 2^{ème} position, par $60 \times 60 (=3600)$ en 3^{ème} position, etc.

Ainsi 75 s'écrit sur deux positions $\vee$; $<\vee\vee\vee\vee\vee$ (60+15) et 200 s'écrit $\vee\vee\vee$; $<<$ ($3 \times 60 + 20$).

☞ Comment écrit-on les nombres 61, 3601, 100 et 2018 en babylonien ?

c) Ce système a été conservé dans notre division du temps en heures, minutes et secondes.

☞ Expliquer cela en donnant un ou deux exemples.

d) Conclusion

☞ Donner au moins un avantage et un inconvénient de ce système de numération ?

1) La numération binaire

a) Les ordinateurs utilisent le *binaire* : deux chiffres (0 et 1) suffisent pour écrire tous les nombres.

On change de position à 2 (car il n'y a pas de chiffre 2), $4=2 \times 2$, $8=2 \times 2 \times 2$, etc.

Ainsi les premiers nombres sont 1, 10 (2+0), 11 (2+1), 100 (4+0), 101 (4+1), 110 (4+2), 111 (4+2+1), ..., etc.

☞ Écrire les nombres de 1 à 20 en binaire.

b) Les nombres 1, 2, 2×2 , $2 \times 2 \times 2$, etc. sont appelés « puissances de 2 » ; on les note 2^0 , 2^1 , 2^2 , 2^3 , etc.

Les puissances de 2 donnent la valeur numérique des chiffres 1 de la notation binaire, selon leur position.

Ainsi 50 s'écrit **110010** car $50=1 \times 2^5 + 1 \times 2^4 + (0 \times 2^3 + 0 \times 2^2) + 1 \times 2^1 + (0 \times 2^0)$

☞ Écrire les dix premières puissances de 2 en notation décimale et en binaire (faire un tableau).

c) La notation binaire indique donc les puissances de 2 employées dans la décomposition d'un nombre.

Ainsi, comme $70=64+4+2=2^6+2^2+2^1$ et que 64, on note ce nombre 1000110 en binaire.

☞ Comment écrit-on 61, 3601, 100 et 2022 en binaire ?

d) En informatique, on appelle « octet », une séquence de 8 chiffres binaires (on dit 8 bits).

Ainsi, le nombre 91 qui s'écrit 01011011 en binaire tient sur un octet.

☞ Combien de nombres différents peut-on coder avec un octet ?

☞ Combien de nombres peut-on coder sur deux octets ?

e) Table ASCII

☞ Quelle fonction peut avoir le tableau ci-contre qui date de 1972 ? (Chercher éventuellement sur internet)

Je m'en suis servi pour écrire le message ci-dessous.

☞ Décoder ce message et répondez-moi en binaire.

01010001-01110101-01100101-01101100-00100000-
01100101-01110011-01110100-00100000-01110100-
01101111-01101110-00100000-01101110-01101111-
01101101-00111111

Pour information :

les bits d'un octet sont numérotés, de droite à gauche : b_1 (unités) à b_7 ($\times 64$) ;

j'ai ajouté un bit nul en position b_8 pour faire un octet.

USASCII code chart

Bits								Column							
b ₈	b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	0	1	2	3	4	5	6	7
								0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	0	0	0	0	1	SOH	DC1	!	1	A	Q	a	q
0	0	0	0	0	0	1	1	STX	DC2	"	2	B	R	b	r
0	0	0	0	1	0	0	0	ETX	DC3	#	3	C	S	c	s
0	0	0	0	1	0	0	1	EOT	DC4	\$	4	D	T	d	t
0	0	0	0	1	0	1	0	ENQ	NAK	%	5	E	U	e	u
0	0	0	0	1	0	1	1	ACK	SYN	&	6	F	V	f	v
0	0	0	1	0	0	0	0	BEL	ETB	'	7	G	W	g	w
0	0	0	1	0	0	0	1	BS	CAN	(	8	H	X	h	x
0	0	0	1	0	0	1	0	HT	EM	)	9	I	Y	i	y
0	0	0	1	0	0	1	1	LF	SUB	*	:	J	Z	j	z
0	0	0	1	0	1	0	0	VT	ESC	+	;	K	[	k	{
0	0	0	1	0	1	0	1	FF	FS	,	<	L	\	l	/
0	0	0	1	0	1	1	0	CR	GS	-	=	M	]	m	}
0	0	0	1	0	1	1	1	SO	RS	.	>	N	^	n	~
0	0	1	0	0	0	0	0	SI	US	/	?	O	_	o	DEL