

Programmation et algorithmique  
 Feuille de TD N°8  
 Cycle de transition **esiea** 2007-2008

L. Beaudoin      R. Erra      D. Julien      V. Guyot

## 1 Vive les bases !

1. Complétez les tableaux suivants :

système décimal	1	2	3	4	5	6	7	8	9	10
système binaire										
système octal										
système hexadécimal										

système décimal	11	12	13	14	15	16	17	18	19	20
système binaire										
système octal										
système hexadécimal										

2. Complétez le tableau suivant :

système binaire	système octal	système hexadécimal	système décimal
1101 1001			
1111 1111			
1010 1010			
0101 0101			
	307		
	377		
		FF	
		AB	

3. complétez le tableau suivant en utilisant la méthode de Horner puis en vérifiant le résultat par la méthode des puissances de 2 :

système décimal	système binaire
1024	
2007	
2812	
3907	

4. On supposera que les entiers relatifs sont codés en binaires sur 8 bits. Complétez le tableau suivant :

entier relatif	valeur absolue binaire	complément à 1	complément à 2
-1			
-32			
-97			
-112			
-128			
77			
101			

5. On supposera que les entiers relatifs sont codés en binaires sur 8 bits. Complétez le tableau suivant

complément à 2	complément à 1	valeur absolue binaire	entier relatif
0000 0100			
1000 0100			
1111 1100			
0101 0101			
1111 1111			
0000 0000			
1001 0110			

6. Un réel en machine est représenté par le triplé  $(s,m,e)$  où  $s$  est le signe du réel,  $m$  sa mantisse et  $e$  l'exposant. Complétez le tableau suivant :

système binaire	système décimal
(0,11.11101,101)	
(1,11.11101,101)	
(1,1011.1011,1001)	
(0,111.010101,101)	

7. On prendra comme version simplifiée de la norme IEEE 754 la norme :

- 1 bit pour le signe,
- 10 bits pour la mantisse,
- 6 bits pour l'exposant biaisé.

Complétez le tableau suivant :

système binaire	système décimal
	0.1
	0.2
	7.75
	9.3
(1,0110000000,110111)	

## 2 Un peu de lecture

### 2.1 Tout ça pour quoi ? Épisode 1

Soit le programme :

```
#include <stdio.h>

int main()
{
    char a,b,c;
    a=127;
    b=a+1;
    c=a+b;
    printf("%d %d %d\n",a,b,c);
    return(0);
}
```

Qu'affiche le programme ? Répondre sur papier puis écrivez le programme pour vérifier si vos réponses sont correctes.

### 2.2 Tout ça pour quoi ? Épisode 2

Soit le programme :

```
#include <stdio.h>
/*****/
int Puissance(int n, int exposant)
{
    int res=1;
    while(exposant!=0)
    {
        res=res*n;
        exposant=exposant-1;
    }
    return(res);
}
/*****/
int main()
{
    int a,b,c,d;
    d=16;
    a=Puissance(2,2*d-1);
    b=Puissance(2,2*d-1)-1;
    c=a+b+2;
    printf("%d\n",c);
    return(0);
}
```

Qu'affiche le programme ? Répondre sur papier puis écrivez le programme pour vérifier si vos réponses sont correctes. On vous assure que la calculatrice n'est absolument pas indispensable !

## 2.3 Tout ça pour quoi ? Épisode 3

Soit le programme suivant :

```
#include <stdio.h>

int main()
{
    unsigned char a=548;
    char b=320;
    printf("%d %c\n",a,a);
    printf("%d %c\n",b,b);
    printf("%d %c\n",a+b,a+b);
    return(0);
}
```

Qu'affiche à l'écran le programme ?

On s'aidera pour répondre de la table ASCII suivante :

Dec	Car	Dec	Car	Dec	Car	Dec	Car	Dec	Car	Dec	Car	Dec	Car
0	\0	16	dle	32	space	48	0	64	@	80	P	96	'
1	soh	17	dc1	33	!	49	1	65	A	81	Q	97	a
2	stx	18	dc2	34	"	50	2	66	B	82	R	98	b
3	etx	19	dc3	35	#	51	3	67	C	83	S	99	c
4	eot	20	dc4	36	\$	52	4	68	D	84	T	100	d
5	enq	21	nak	37	%	53	5	69	E	85	U	101	e
6	ack	22	syn	38	&	54	6	70	F	86	V	102	f
7	\a	23	etb	39	'	55	7	71	G	87	W	103	g
8	\b	24	can	40	(	56	8	72	H	88	X	104	h
9	\t	25	em	41	)	57	9	73	I	89	Y	105	i
10	\n	26	sub	42	*	58	:	74	J	90	Z	106	j
11	\v	27	esc	43	+	59	;	75	K	91	[	107	k
12	\f	28	fs	44	,	60	<	76	L	92	\	108	l
13	\r	29	gs	45	-	61	=	77	M	93	]	109	m
14	so	30	rs	46	.	62	>	78	N	94	^	110	n
15	si	31	us	47	/	63	?	79	O	95	_	111	o
												112	p
												113	q
												114	r
												115	s
												116	t
												117	u
												118	v
												119	w
												120	x
												121	y
												122	z
												123	{
												124	
												125	}
												126	~
												127	del

### 3 L'explosion d'Ariane 5

Le premier tir d'Ariane 5 s'est soldé par un échec au bout de 45 secondes à cause d'un problème informatique... Élémentaire. Bien que le vrai problème soit plus complexe, cet exercice devrait vous faire sentir la nécessité de bien comprendre la représentation des données en machine.

1. Écrire la fonction `Explosion` qui affiche à l'écran `BOUM!!!` lorsque l'altitude `Alti` perçue par la fusée (codée en `unsigned char` pour prendre moins de place mémoire) est nulle. Le prototype de la fonction est :  
`void Explosion(unsigned char Alti)`
2. Écrire la fonction `Montee` qui affiche à l'écran par pas de 1 km entre 1 et 300 km l'altitude réelle actuelle de la fusée (codée en `int`) et l'altitude perçue par la fusée (codée en `unsigned char`). Bien entendu, pour chaque altitude, la fusée teste son altitude avec la fonction `Explosion`. Le prototype de `Montee` est :  
`void Montee()`

Pourquoi ce résultat ?

### 4 Bases et opérations arithmétiques élémentaires

Effectuez les opérations suivantes dans les bases demandées (l'indice en bas à droite de chaque nombre).

$$\begin{array}{r} (101010)_2 \\ + (101010)_2 \\ \hline \end{array}$$

$$\begin{array}{r} (77777)_8 \\ + (12345)_8 \\ \hline \end{array}$$

$$\begin{array}{r} (ABC)_{16} \\ + (CBA)_{16} \\ \hline \end{array}$$

$$\begin{array}{r} (101010)_2 \\ - (10101)_2 \\ \hline \end{array}$$

$$\begin{array}{r} (77777)_8 \\ - (12345)_8 \\ \hline \end{array}$$

$$\begin{array}{r} (CBA)_{16} \\ - (ABC)_{16} \\ \hline \end{array}$$

$$\begin{array}{r} (101010)_2 \\ * (101)_2 \\ \hline \end{array}$$

$$\begin{array}{r} (1234)_8 \\ * (123)_8 \\ \hline \end{array}$$

$$\begin{array}{r} (ABC)_{16} \\ * (123)_{16} \\ \hline \end{array}$$