

Comment sont représentés les entiers négatifs en mémoire ?

1 Addition de deux nombres binaires

Une addition en base 2 applique les mêmes principes qu'en base 10 :

- $0 + 0 = 0$
- $1 + 0 = 1$
- $1 + 1 = 0$ et une retenue de 1
- $1 + 1 + 1 = 1$ et une retenue de 1

$$\begin{array}{r}
 \\
 \\
 + \\
 \hline

 \end{array}$$

2 Une représentation naïve des entiers négatifs

2.1 Bit de poids fort

Une première idée serait d'utiliser ce bit comme marqueur de signe :

- 0 pour un entier positif,
- 1 pour un entier négatif.

Ainsi l'entier -25 serait encodé dans un mot mémoire de 1 octet :

$$-25_{10} = 10011001_2$$

2.2 Inconvénients de la représentation

- Il y a 2 zéros.
- L'algorithme d'addition ne fonctionne pas.

3 Le complément à 2 puissance n

3.1 Définition

La valeur $2^n - |x|$ représente l'entier négatif x .

1	1	1	1	1	1	1	1	=	-1	$2^8 - -1 = 255$
1	1	1	1	1	1	1	0	=	-2	$2^8 - -2 = 254$
1	...							=	...	
1	0	0	0	0	0	0	1	=	-127	$2^8 - -127 = 129$
1	0	0	0	0	0	0	0	=	-128	$2^8 - -128 = 128$
0	1	1	1	1	1	1	1	=	127	
0	...							=	...	
0	0	0	0	0	0	1	0	=	2	
0	0	0	0	0	0	0	1	=	1	
0	0	0	0	0	0	0	0	=	0	

3.2 Calculer le complément à 2

Pour coder (-20) :

1. — Prendre le nombre positif 20 : 00010100
 - Inverser les bits : 11101011
 - Ajouter 1 : 11101100
 - -20 : 11101100
2. — Prendre le nombre positif 20 : 00010100
 - Garder la partie à droite telle quelle : 00010100
 - Inverser la partie de gauche après le premier un : 11101100
 - -20 : 11101100

3.3 Intérêt de la méthode

- Il n'y a qu'un seul zéro.
- On peut appliquer l'algorithme d'addition.