Représentation des entiers relatifs

Représentation des entiers relatifs

Christophe Viroulaud

Première - NSI

DonRep 02

Représentation des entiers relatifs

Christophe Viroulaud

Première - NSI

DonRep 02

Représentation des entiers relatifs

Addition de deux nombres binaires

naive des entiers négatifs Bit de poids fort

> complément à 2 uissance *n*

finition Ilculer le complément à 2 térêt de la méthode



Un système 64 bits peut représenter 2^{64} entiers.

```
1 >>> import sys
2 >>> sys.maxsize
3 9223372036854775807
```

Code 1 – Cette valeur correspond à $2^{63} - 1$.

Observation

Un des bits ne semble pas utilisé.

Représentation des entiers relatifs

Addition de deux nombres binaires

aïve des entiers égatifs

Inconvénients de la représentation

e complément à 2 iissance *n*

finition Ilculer le complément à 2

térêt de la méthode

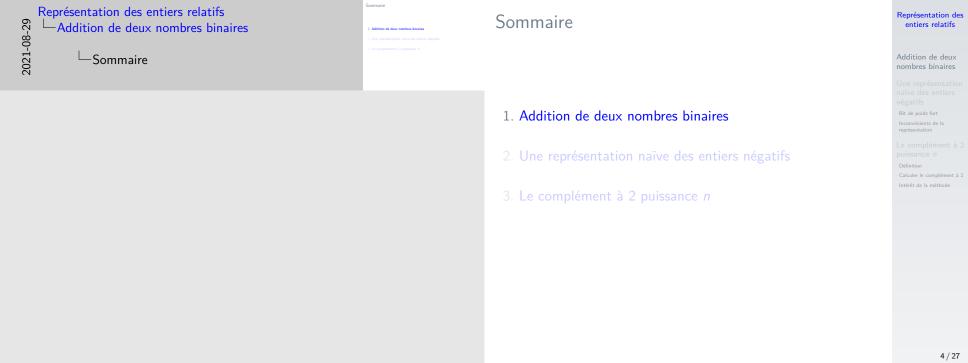
n mémoire?

Comment sont représentés les entiers négatifs en mémoire?

Représentation des entiers relatifs

Bit de poids fort

Intérêt de la méthode



▶ 0 + 0 = 0 ▶ 1+0=1

► 1+1+1=1 et une retenue de 1

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

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

D'après l'intro dans 8 bits on peut stocker des nombres allant jusqu'à

$$2^7 - 1 = 127_{10} = 01111111$$

 $2^7 - 1 = 127_{10} = 011111111_2$

Dans un mot mémoire de 1 octet :

 $25_{10} = 00011001_2$

nombres binaires

Représentation des

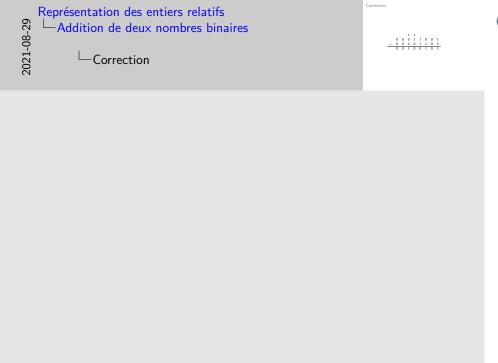
entiers relatifs

Addition de deux

3. Convertir le résultat en base 10. Le résultat est-il

Activité 1 :

- 1. Convertir 25 et 12 en base 2.
- 2. Effectuer l'addition binaire de ces nombres.
- 3. Convertir le résultat en base 10. Le résultat est-il correct?



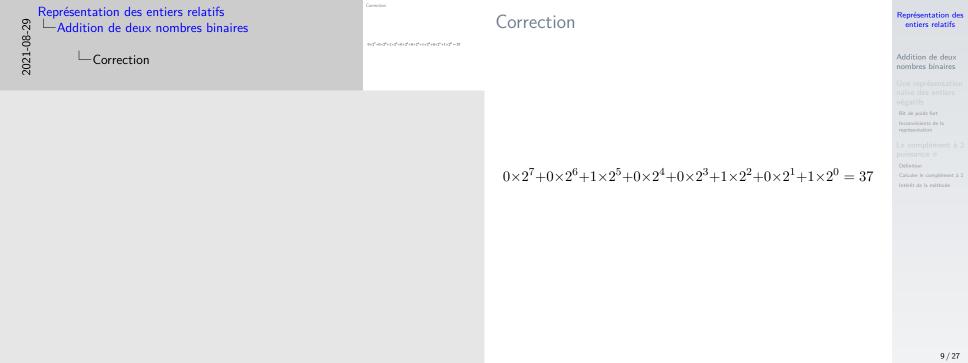
Correction

Addition de deux nombres binaires

Représentation des

entiers relatifs

Bit de poids fort





Bit de poids fort

Le bit le plus à gauche de la représentation n'est pour l'instant pas utilisé. C'est le **bit de poids fort**.

À retenir

Pour représenter un nombre entier, il faut connaître la taille du mot mémoire.

Représentation des entiers relatifs

Addition de deux nombres binaires

Une représentation naïve des entiers négatifs

Bit de poids fort Inconvénients de la

> e complément à 2 uissance *n*

éfinition alculer le complément à 2

Représentation des entiers relatifs

Une représentation naïve des entiers négatifs

Bit de poids fort

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

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

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

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 :

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

Addition de deux

Représentation des entiers relatifs

Une représentation naïve des entiers négatifs

Bit de poids fort

complément à 2 issance *n*

éfinition alculer le complément à 2 itérêt de la méthode



Sommaire

- 1. Addition de deux nombres binaires
- 2. Une représentation naı̈ve des entiers négatifs
- 2.1 Bit de poids fort
- 2.2 Inconvénients de la représentation
- 3 Le complément à 2 puissance n

Représentation des

entiers relatifs

Inconvénients de la représentation



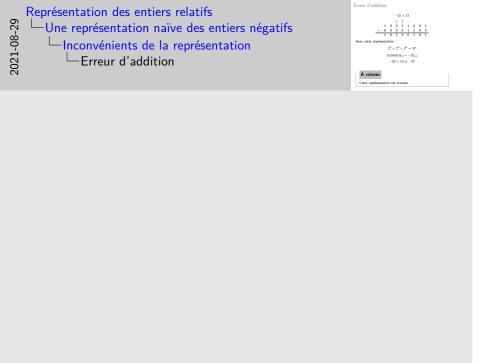
Le zéro

représentation Dans un système 8 bits le zéro est représenté par 00000000_2 . Cependant 10000000_2 se traduit par -0. Il y a donc deux représentations pour zéro.

Représentation des

entiers relatifs

Inconvénients de la



Erreur d'addition

 $2^5 + 2^2 + 2^0 = 37$

Avec cette représentation :

$$10100101_2 = -37_{10}$$
$$-25 + 12 \neq -37$$

À retenir

Cette représentation est erronée.

ne représentation ive des entiers

Représentation des

entiers relatifs

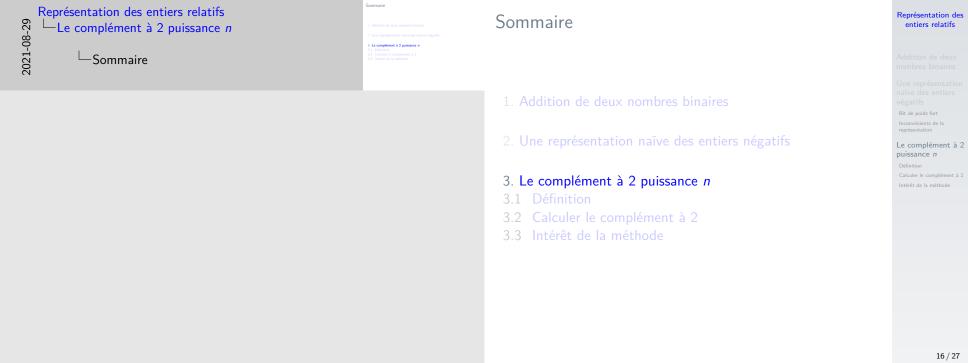
Inconvénients de la représentation

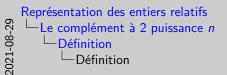
Le complément à

Duissance *n*Définition

Calculer le complément

culer le complément à rêt de la méthode







Définition

Le complément à 2 puissance n est une représentation qui ne change rien pour les entiers positifs. Ainsi sur 8 bits :

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

Représentation des entiers relatifs

Addition de deux nombres binaires

Une représentation naïve des entiers négatifs

Bit de poids fort Inconvénients de la

> e complément à 2 uissance *n*

Définition

Calculer le complément à 2 Intérêt de la méthode Par contre la valeur $2^a - |x|$ représente l'entier négatif x. Ainsi sur 8 bits, -1 s'écrit $2^6 - 1 = 256 - 1 = 255 = 11111111_0$

										-
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	$2^8 - -127 = 12$
1	0	0	0	0	0	0	0	-	-128	$2^8 - -128 = 12$
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	

Par contre la valeur $2^n - |x|$ représente l'entier négatif x. Ainsi sur 8 bits, -1 s'écrit

$$2^8 - 1 = 256 - 1 = 255_{10} = 11111111_2$$

Addition de deux nombres binaires Une représentation

Représentation des

entiers relatifs

naive des entiers négatifs Bit de poids fort

1	1	1	=	-1	$ 2^8- -1 =255$ replésentatio
1	1	0	=	-2	$ 2^8 - -2 = 254$ Le complement to the complement of the comple
			=		Définition Calculer le o
0	0	1	=	-127	$\mid 2^8 - \mid -127 ert = 129$ Interêt de la
0	0	0	=	-128	$2^8 - -128 = 128$
1	1	1	=	127	
			=		
0	1	0	=	2	
0	0	1	=	1	
0	0	0	=	0	



Sommaire

- 1 Addition do douv nombres binaires
- 2. Una raprésantation païva des antiers négatifs
- 3. Le complément à 2 puissance *n*
 - finition
 - efinition
- 3.2 Calculer le complément à 2
 - e compleme

Représentation des

entiers relatifs

Calculer le complément à 2

Pour coder (-20):

Calculer le complément à 2

Pour coder (-20) : ► Prendre le nombre positif 20 : 00010100

► -20:11101100

- ▶ Prendre le nombre positif 20 : 00010100
- ► Inverser les bits : 11101011
- ► Ajouter 1 : 11101100
- -20:11101100

Représentation des

entiers relatifs

Calculer le complément à 2

Seconde méthode

(compris) puis d'inverser tous les suivants.

Inverser la partie de gauche après le premier un
 11101100
 → 20 : 11101100

Garder tous les chiffres depuis la droite jusqu'au premier 1 (compris) puis d'inverser tous les suivants.

- ► Prendre le nombre positif 20 : 00010100
- ► Garder la partie à droite telle quelle : 00010<u>100</u>
- ► Inverser la partie de gauche après le premier un : 11101100
- ► -20 : 11101100

2021-08-29

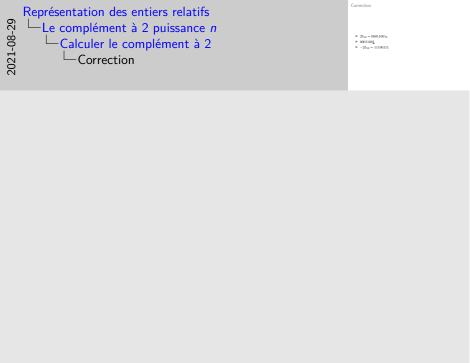
Activité 2 : Calculer le complément à 2 (sur 1 octet) de -25.

Activité 2 : Calculer le complément à 2 (sur 1 octet) de -25.

Représentation des

entiers relatifs

Calculer le complément à 2



Correction

- \triangleright 25₁₀ = 00011001₂
- **▶** 00011001
- $-25_{10} = 11100111$

23 / 27

Représentation des

entiers relatifs

Calculer le complément à 2



Sommaire

- Addition do doux nombres hinaires
- 2 Une représentation naïve des entiers négatifs
- 3. Le complément à 2 puissance *n*
 - finition
 - finition
 - uler le complément à
- 3.3 Intérêt de la méthode

Représentation des

entiers relatifs

Intérêt de la méthode



Intérêt de la méthode

Il n'y a qu'un seul zéro.

Représentation des

entiers relatifs

Intérêt de la méthode



-25 + 12 = 11100111 + 00001100 = 11110011 = 243 et $243 - 2^8 = 243 - 256 = -13$ $2^7 + 2^6 + 2^5 + 2^4 + 2^1 + 2^0 = 243$ $243 - 2^8 = 243 - 256 = -13$

 $243 - 2^8 = 243 - 256 = -13$

Avec cette représentation :

-25 + 12 = 11100111 + 00001100 = 11110011 = 243 et

-25 + 12

 $2^7 + 2^6 + 2^5 + 2^4 + 2^1 + 2^0 = 243$

 $243 - 2^8 = 243 - 256 = -13$

Représentation des

entiers relatifs

Intérêt de la méthode

Addition

À retenir

Les nombres entiers négatifs sont représentés par le complément à 2.

Représentation des

entiers relatifs

Intérêt de la méthode