Université Mohammed V
Faculté des Sciences Rabat
Département d'informatique

Filière SMIA
Module M6
Initiation à l'Informatique

Correction des Séries de TD N° 3 et N° 4

Correction de la Série 3 :

Correction de l'exercice 1 :

décimale	octale	hexadécimale	binaire
1016	1770	3F8	1111111000
2100	4064	834	100000110100
2730	5252	AAA	101010101010
292	444	124	100100100

Correction de l'exercice 2 :

Donnez les représentations binaires sur 8 bits de -115 en utilisant les trois représentations :

1. Binaire signé : avec 8 bits on peux coder les nombres de -127 à 127.
   115 = 112 + 3 = 7*16 + 3 = 01110000_2+ 00000011_2 = $01110011_{(bp)}$ (en binaire pure)
   et -115 = $11110011_{(bs)}$ en binaire signé.

2. Complément à 1 : avec 8 bits on peux coder les nombres de -127 à 127.
   115 = $01110011_{(bp)}$ (en binaire pure)
   - 115 = $10001100_{(cà1)}$ (en complément à 1)

3. Complément à 2 : avec 8 bits on peux coder les nombres de -128 à 127.
   - 115 = $10001100_{(cà1)}$ (en complément à 1), donc - 115 = $10001100_{(cà1)}$ + 1 = $10001101_{(cà2)}$.

Correction de l'exercice 2 :

Donnez la valeur en base 10 des nombres binaires 01010101, 10010001, selon que l'on les lit en considérant un codage d'entiers sur 8 bits :

1. En binaire pur : $01010101_{(bp)}$= 85 $10010001_{(bp)}$ = 145
2. En binaire signé : $01010101_{(bs)}$ = 85 $10010001_{(bs)}$ = -17
3. En complément à 1 : $01010101_{(cà1)}$= 85 $10010001_{(cà1)}$ = -110
4. En complément à 2 : $01010101_{(cà2)}$= 85 $10010001_{(cà2)}$ = -111

Correction de l'exercice 3 :

Additionnez en binaire -115 et 92, puis -115 et -2 dans les deux représentations complément à 1 et complément à 2 :

-115 + 92 = $10001100_{(cà1)}$ + $01011100_{(cà1)}$ = $11101000_{(cà1)}$ = -23 ok
-115 - 2 = $10001100_{(cà1)}$ + $11111101_{(cà1)}$ = $10001001_{(cà1)}$ = -118 non (Complément à 1 ne marche pas !).
-115 + 92 = $10001101_{(cà2)}$ + $01011100_{(cà2)}$ = $11101001_{(cà2)}$ = -23 ok
-115 - 2 = $10001101_{(cà2)}$ + $11111110_{(cà2)}$ = $10001011_{(cà2)}$ = -117 ok

Correction de la Série 4 :

Correction de l'exercice 1 :

1. **a**. la 1ère bit est 1 donc le nombre est négatif. Les 8 bits suivants 10000010_2=130, donc E_b=130-127=3. La mantisse M = 11110110000…0. Donc le nombre est -1,11110110 * 2^3 = -1111,10110 = **-15,6875**

b. la 1ère bit est 0 donc le nombre est positif. Les 8 bits suivants 10000001_2=129, donc E_b=129-127=2. La mantisse M = 1110000…0. Donc le nombre est 1,1110 * 2^2 = 111,10 = **7,5**

c. la 1ère bit est 1 donc le nombre est négatif. Les 8 bits suivants 10000100_2=132, donc E_b=132-127=5. La mantisse M = 0001110000…0. Donc le nombre est -1,0001110 * 2^5 = -100011,10 = **-35,5**

2.

a. Le nombre est négatif, donc la première bit du codage est 1.
-123,75 =-1111011,11 car 123=120+3=15*8+3= 1111000_2 +11_2 = 01111011_2, et 0,75*2=1,5 d'où le premier 1 après la virgule, ensuite 0,5*2=1,0 ce qui donne le dernier 1.
-123,75 = -1,11101111 * 2^6. Donc la mantisse M est 1110111100…0 et E_B=127+6=133=10000101_2. Donc le codage de -123,75 est
1 10000101 11101111000000000000000.

b. Le nombre est positif, donc la première bit du codage est 0.
6,125 = 110,001 car 6=110_2, et 0,125*2=0,25 d'où le premier 0 après la virgule, ensuite 0,25*2=0,5 d'où le deuxième 0 après la virgule, ensuite 0,5*2=1,0 ce qui donne le dernier 1.
6,125 = 1,10001 * 2^2. Donc la mantisse M est 1000100…0 et E_B=127+2=129=10000001_2. Donc le codage de 6,125 est
0 10000001 10001000000000000000000.

Correction de l'exercice 2 :

1. Le code de la lettre 'o' est 111_{10} = 96+15 = 1100000_2+1111_2=**01101111_2**.
2. Le code de la lettre 'n' est 01101110_2 = 111_{10} -1 = **110_{10}** .
3. En mémoire, on trouve la séquence suivante 01101010 10010101.
   - **a.** Pour "deux nombres entiers naturels (sur 8 bits) codés en Binaire Pure", on trouve :
     01101010_2 = 1100000_2 + 1010_2 = 96+10=**106**.
     10010101_2 = 10000000_2 + 10101_2 = 128+21=**149**.
   - **b.** Pour "deux nombres entiers relatifs sur 8 bits en représentation complément à 1", on trouve :
     Le nombre est positif : $01101010_{(c\ à\ 1)}$ = 96+10=**106**.
     Le nombre est négatif : En inversant les bits, on a 01101010_2 = 106. Donc le nombre $10010101_{(c\ à\ 1)}$ = **-106**.
   - **c.** Pour "deux nombres entiers relatifs sur 8 bits en représentation complément à 2", on trouve :
     Le nombre est positif : $01101010_{(c\ à\ 2)}$ = 96+10=**106**.
     Le nombre est négatif : En inversant les bits, on a 01101010_2 et on ajoute 1 on trouve 01101011_2 =-107. Donc le nombre $10010101_{(c\ à\ 2)}$ = **-107**.
   - **d.** Pour "deux caractères du code ASCII étendu", on trouve :
     01101010_2 =106 qui le carartère '**j**'.
     10010101_2 =149 qui est le caratère '**Ò**'