

Exercice 1 : Quelques généralités

- (a) Comment écrire en C une constante littérale hexadécimale correspondant à la valeur $1ab4_{16}$?

| 0x1ab4

- (b) Comment écrire en C une constante littérale octale représentant la valeur 2642_8 ?

| 02642

- (c) Bien qu'il ne soit pas normalisé dans C11, le préfixe **0b** est utilisé pour représenter une constante littérale binaire.

- (d) Le signe d'un nombre en complément à deux peut être connu en observant son bit de poids **fort** tandis que le bit de poids **faible** renseigne sur sa parité (pair ou impair).

- (e) Le programme suivant est appelé. Combien d'arguments le programme reçoit-il ?

```
./a.out 1 2 3
```

| 4 arguments

- (f) Comment s'appelle le flux qui permet à un programme de recevoir des informations par exemple saisies au clavier ?

| L'entrée standard

- (g) Combien de bytes y a-t-il dans un Mebibyte (MiB) ?

| $1024 \cdot 1024 = 1'048'576$ Bytes

- (h) Combien de bits faut-il pour représenter un Gigabyte (GB) de données ?

| $\lceil \log_2(1 \cdot 10^9) \rceil \approx \lceil 29.89 \rceil = 30$ bits

Exercice 2 : Effectuer les additions suivantes à la main en binaire sur 8 bits

Exemple :

```

      11
00001001 = 0x09 = 9
+ 00011000 = 0x18 = 24
-----
00100001 = 0x21 = 33

```

(a) $1 + 51$

```

      11
00000001 = 0x01 = 1
+ 00110011 = 0x33 = 51
-----
00110100 = 0x34 = 52

```

(b) $51 - 7$

```

1111 11
00110011 = 0x33 = 51
+ 11111001 = (~00000111 + 1) = 0xF9 = -7
-----
00101100 = 0x2C = 44

```

(c) $204 + 51$

```

11001100 = 0xCC = 204
+ 00110011 = 0x33 = 51
-----
11111111 = 0xFF = 255

```

(d) $204 + 204$

```

      11
11001100 = 0xCC = 204
+ 11001100 = 0xCC = 204
-----
10011000 = 0x98 = (408-256) = 152

```

Exercice 3: Programmation

- (a) Écrire un programme qui lit le premier nombre passé en argument depuis la ligne de commande et l'affiche à l'écran.

```
int main(int argc, char *argv[]) {  
    printf("Votre nombre est : %d\n", atoi(argv[1]));  
}
```

- (b) Écrire un programme qui affiche explicitement la somme de deux nombres. Ces deux nombres sont déclarés dans des variables entières nommées a et b.
Voici ce que produit le programme :

```
./a.out  
42 + 23 = 65
```

```
int main() {  
    int a = 42;  
    int b = 23;  
    printf("%d + %d = %d", a, b, a + b);  
}
```