

Exercice 1. Conversions et opérations en base 2

Question 1 – On a :

$$\begin{aligned}(1\ 0000\ 0000)_2 &= 2^8 = 256 \\ -(1\ 1011)_2 &= -(1 + 2 + 8 + 16) = -27 \\ (10\ 0100\ 0100)_2 &= 4 + 64 + 512 = 580\end{aligned}$$

Pour le vérifier en Python :

```
print(0b100000000)
print(-0b11011)
print(0b1001000100)
```

Question 2 – On a :

$$\begin{aligned}103 &= 64 + 32 + 4 + 2 + 1 = (110\ 0111)_2 \\ 600 &= 512 + 64 + 16 + 8 = (10\ 0101\ 1000)_2 \\ -255 &= -(256 - 1) = -((1\ 0000\ 0000)_2 - (1)_2) = -(1111\ 1111)_2\end{aligned}$$

Pour le vérifier en Python :

```
print(bin(103))
print(bin(600))
print(bin(-255))
```

Question 3 – On a :

$$\begin{array}{rcccc} & 1 & 0 & 1 & 0 \\ + & 1 & 1 & 0 & 0 \\ \hline 1 & 0 & 1 & 1 & 0 \end{array} \qquad \begin{array}{rcccccc} & & & 1 & 0 & 1 & 1 & 1 \\ + & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{array}$$

Donc $(1010)_2 + (1100)_2 = (1\ 0110)_2$ et $(1\ 0111)_2 + (1011\ 0011)_2 = (1100\ 1010)_2$.

$$\begin{array}{rcccc} & 1 & 1 & 0 & 0 \\ - & & 1 & 1 & 1 \\ \hline 0 & 1 & 0 & 1 \end{array} \qquad \begin{array}{rcccc} & 1 & 1 & 0 & 0 \\ - & & 1 & 1 & 0 \\ \hline 0 & 1 & 1 & 0 \end{array}$$

Donc $(1100)_2 - (111)_2 = (101)_2$ et $(110)_2 - (1100)_2 = -(110)_2$.

$$\begin{array}{rcccccc} & & & 1 & 1 & 1 & 0 & 1 \\ \times & & & 1 & 1 & 1 & 0 \\ \hline & & & 1 & 1 & 1 & 0 & 1 & 0 \\ + & & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ + & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ \hline & & 1 & 1 & 1 & 0 & 1 & 0 \\ + & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{array}$$

Donc $(1\ 1101)_2 \times (1110)_2 = (1\ 1001\ 0110)_2$.

$$\begin{array}{rcccc|cccc} & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & & 1 & 1 & 1 & 0 \\ - & & 1 & 1 & 1 & 0 & & & & & 1 & 0 & 1 & 1 \\ \hline & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & & & & & \\ - & & & & 1 & 1 & 1 & 0 & & & & & & \\ \hline & & & 0 & 0 & 1 & 1 & 1 & 1 & & & & & \\ - & & & & & 1 & 1 & 1 & 0 & & & & & \\ \hline & & & & & 0 & 0 & 0 & 1 & & & & & \end{array}$$

Donc $(1001\ 1011)_2 // (1110)_2 = (1011)_2$ et $(1001\ 1011)_2 \% (1110)_2 = (1)_2$.
Pour le vérifier en Python :

```
print(bin(0b1010 + 0b1100))
print(bin(0b10111 + 0b10110011))
print(bin(0b1100 - 0b111))
print(bin(0b110 - 0b1100))
print(bin(0b11101*0b1110))
print(bin(0b10011011//0b1110))
print(bin(0b10011011%0b1110))
```