

## 01.Calcul en Python

Nous utiliserons dans ce TP le langage de programmation *Python* au moyen de l'environnement de développement *IDLE*.

### 1 Calcul avec les flottants

Le symbole d'affectation en Python est =, les opérateurs +, -, \*, / et \*\* permettent de calculer somme, différence, produit, quotient et puissance de flottants.

```
>>> x=2
>>> y=8
>>> x/y
0.25
>>> x**y
256
```

**Exercice 1.** Déterminer l'entier naturel  $n$  et le nombre réel  $m \in [1, 2[$  tels que  $1696 = m \times 2^n$ .

### 2 Calcul avec les entiers

Les opérateurs // et % permettent de calculer quotient et reste de la division euclidienne de deux entiers.

```
>>> a,b=25,8
>>> a//b
3
>>> a%b
1
```

**Exercice 2.** Déterminer l'écriture binaire de l'entier 1696.

### 3 Calcul avec les booléens

Les opérateurs de comparaison en Python sont ==, !=, <, >, <= et >=.

```
>>>2==3
False
>>>2!=3
True
```

**Exercice 3.** Dans quels cas la valeur de l'expression  $e \% 2 == 1$  est-elle True ou False pour  $e$  une variable de type entier ?

Les opérateurs logiques en Python sont and, or et not.

```
>>>not(True) or False
False
```

**Exercice 4.** Déterminer la table de vérité de  $\text{NON}(A) \text{ ET } B$ .

## 4 Opérations sur les chaînes de caractères

Les opérateurs de concaténation sur les chaînes de caractères sont + et \*.

```
>>> a="bon"
>>> b="jour"
>>> (a+b)*2
'bonjourbonjour'
```

## 5 Opérateurs bit à bit

**Exercice 5.** Expliquer le fonctionnement des opérateurs bit à bit &, | et ^.

```
>>> p,q=12,37
>>> p&q
4
>>> p|q
45
>>> p^q
41
```

## Réponses

1) On a  $1696 = 1,65625 \times 2^{10}$

```
>>> 1696/2**10
1.65625
```

2) L'écriture binaire de 1696 est 11010100000.

```
>>> 1696%2,1696//2
(0, 848)
>>> 848%2,848//2
(0, 424)
>>> 424%2,424//2
(0, 212)
>>> 212%2,212//2
(0, 106)
>>> 106%2,106//2
(0, 53)
>>> 53%2,53//2
(1, 26)
>>> 26%2,26//2
(0, 13)
>>> 13%2,13//2
(1, 6)
>>> 6%2,6//2
(0, 3)
>>> 3%2,3//2
(1, 1)
>>> 1%2,1//2
(1, 0)
```

3) True pour  $e$  impair et False pour  $e$  pair.

```
>>> e=7
>>> e%2==1
True
```

A	B	NON(A) ET B
F	F	F
F	V	V
V	F	F
V	V	F

4)

```
>>> A,B=False,False
>>> not(A) and B
False
>>> A,B=False,True
>>> not(A) and B
True
>>> A,B=True,False
>>> not(A) and B
False
>>> A,B=True,True
>>> not(A) and B
False
```

5) Les opérateurs  $\&$ ,  $|$  et  $\wedge$  correspondent respectivement au *ET*, *OU inclusif* et *OU exclusif*.

décimal	binaire
12	001100
37	100101
4	000100
45	101101
41	101001