

# Doc n°2: priorité des opérations en Python

En Python, les règles de priorité sont les mêmes que celles des mathématiques. On les mémorise à l'aide de l'acronyme **PEMDAS** :

- **P** pour *parenthèses*.

Ce sont elles qui ont la plus haute priorité. Elles permettent donc de « forcer » l'évaluation d'une expression dans l'ordre désiré.

Ainsi  $2*(3-1) = 4$ , et  $(1+1)**(5-2) = 8$ .

- **E** pour *exposants*.

Les puissances sont évaluées ensuite (après les expressions entre parenthèses), avant les autres opérations.

Ainsi  $2**1+1 = 3$  (et non 4), et  $3*1**10 = 3$  (et non 59049!).

- **M** et **D** pour *multiplication* et *division*, qui ont la même priorité.
- Elles sont évaluées avant l'*addition A* et la *soustraction S*, lesquelles sont donc effectuées en dernier lieu.

Ainsi  $2*3-1 = 5$  (plutôt que 4), et  $6/3-1 = 1$

- Si deux opérateurs ont la même priorité, l'évaluation est effectuée de gauche à droite.

Ainsi dans l'expression  $59*100//60$ , la multiplication est effectuée en premier, et la machine doit donc ensuite effectuer  $5900//60$ , ce qui donne **98**. Si la division était effectuée en premier, le résultat serait **59** (puisqu'il s'agit d'une division entière).

```
>>> 59*100//60
```

```
98
```

```
>>> 59*(100//60)
```

```
59
```

```
>>> (59*100)//60
```

```
98
```

```
>>> 60/5/4
```

```
3.0
```

```
>>> 60/(5/4)
```

```
48.0
```