

03_conditions-embranchements

November 29, 2023

1 Activité : conditions et embranchements

Adapté de C. Poulmaire / AEIF

2 Exercice 1 : opérateurs de comparaisons et opérateurs logiques

1. Déterminer les résultats de chacun des tests suivants puis vérifier en exécutant les cellules :

```
[ ]: 7 >= 0
[ ]: 5 == 3
[ ]: 3 != 3
[ ]: 7 < 3.8
[ ]: 4 < 4
[ ]: 4 <= 4
[ ]: 4 > -2
[ ]: 5 != 5.1
[ ]: 3.9 == 2.4
[ ]: -3.2 < 4.3
[1]: -3.2 >= 4.3
[1]: False
```

2. Déterminer les résultats de chacun des tests suivants puis vérifier en exécutant les cellules :

```
[ ]: 4 == 4.0
[ ]: 2*0.1 == 0.2
[ ]: 3*0.1 == 0.3
```

```
[ ]: 0.7**2 != 0.49
```

Tout correspond bien aux résultats que vous attendiez...?

3. Déterminer les résultats de chacune des conditions suivantes puis vérifier en exécutant les cellules :

```
[ ]: a = 3
a >= 0 and a < 5
```

```
[ ]: a = 3
0 <= a < 5    # Python autorise le test en double inégalité. Cette cellule est
↪équivalente à la cellule précédente
```

```
[ ]: not(4 > 3)
```

```
[ ]: a = 3
b = 5
a > b and a > 0
```

```
[ ]: a = 3
b = 5
a > b or a > 0
```

```
[ ]: a = 4
b = a + 1
b - a == 1
```

```
[ ]: a = 4
b = a + 1
not(a - b > 0)
```

2.1 Exercice 2: Fonction maximum

```
[1]: # Ecrire ici une fonction max_a_b(a, b) qui prend en arguments deux entiers et
↪renvoie le maximum
```

Insérez ci-dessous plusieurs cellules de code pour y effectuer différents appels de votre fonction; essayez de penser à tous les cas possibles (inspirez-vous de ce que vous avez fait plus tôt avec votre fonction `perimetre_rectangle`).

2.2 Exercice 3: Fonction maximum et minimum

```
[2]: # En recopiant votre fonction précédente, enrichissez-là de manière à ce
↪qu'elle prenne un troisième paramètre
# "maxmin" qui peut avoir la valeur "max" (et alors la fonction renvoie le
↪maximum) ou "min" (le minimum).
```

Comme pour la fonction précédente, insérez ici des cellules de code pour tester cette nouvelle fonction.