

TD 1: CONDITIONNELS

1. Donnez et expliquez le résultat des instructions Python suivantes :

- (a) `float(123)`
- (b) `float('123')`
- (c) `float('123.23')`
- (d) `int(123.23)`
- (e) `int('123.23')`
- (f) `int(float('123.23'))`
- (g) `str(12)`
- (h) `bool('a')`
- (i) `bool(0)`
- (j) `bool(0.1)`

2. Donnez et expliquez le résultat des instructions Python suivantes :

- (a) `1 == 1`
- (b) `1 == True`
- (c) `0 == False`
- (d) `3 == 1 * 3`
- (e) `(3 == 3) * 4 + 3 == 1`
- (f) `3**5 >= 4**4`

3. Sur une calculatrice défectueuse affichant un nombre entier, nous pouvons effectuer deux types d'opérations **dans l'ordre suivant** :

- D'abord, nous pouvons effectuer uniquement des multiplications par 2 (opération “doubler”).
- Ensuite, une fois que nous avons terminé toutes les multiplications, nous pouvons effectuer uniquement des soustractions de 1 (opération “décrémenter”).

Nous ne pouvons **pas** alterner entre ces opérations : toutes les multiplications doivent être effectuées en premier, suivies de toutes les décrémentations.

Initialement, la calculatrice affiche le entier x .

Écrivez un programme Python qui, étant donné deux entiers $x \geq 1$ et $y \geq 0$, calcul le nombre minimum d'opérations nécessaires pour afficher le nombre y , à partir d'entier x .

Vous pouvez utiliser la fonction `math.log2(x)` qui retournera le flottant $\log_2(x)$.

Exemple : Pour $x = 2$ et $y = 9$:

- on fait 3 multiplications : $2 \rightarrow 4 \rightarrow 8 \rightarrow 16$, puis $16 \rightarrow 15 \rightarrow 14 \rightarrow 13 \rightarrow 12 \rightarrow 11 \rightarrow 10 \rightarrow 9$ (7 décrémentations). Total : 3 multiplications + 7 décrémentations = 10 opérations

4. Un boomerang est un ensemble de trois points distincts et non alignés dans le plan.
Soit trois points dans le plan, x_1, y_1 and x_2, y_2 and x_3, y_3 , indiquer si ces points sont un boomerang.
Exemple: Pour `x1=1, y1=1, x2=2, y2=3, x3=3, y3=2`, la solution est **True**.
5. Écrivez un programme Python qui, étant donné trois nombres—`num1, num2, num3`—les trie avec au plus trois *comparaisons*.
Une ‘`if condition: ... else: ...`’ est considérée comme une seule comparaison.
6. Écrivez un programme Python qui, étant donné quatre nombres—`num1, num2, num3, num4`—les trie.
7. Écrivez un programme Python qui, étant donné quatre nombres—`num1, num2, num3, num4`—les trie.
Votre code doit contenir au plus 6 instructions `if`.
8. (**difficile**) Écrivez un programme Python qui, étant donné quatre nombres—`num1, num2, num3, num4`—les trie.
Votre code doit contenir au plus 5 instructions `if`.