

Contrôle 1 : Instructions de base

Corrigé

1. Opérations élémentaires

Indiquer dans le tableau ci-dessous de quelle instruction/fonction/indication il s'agit :

| | | | |
|--------------|---|--------------|---------------------------------------|
| abs | fonction valeur absolue | % | reste dans la division euclidienne |
| type | fonction type de donnée | true | valeur booléenne : vrai |
| float | type nombre à virgule | false | valeur booléenne : faux |
| int | type entier relatif | bool | type booléen (vrai ou faux) |
| int | fonction partie entière | == | test "est égal" |
| * | multiplication | != | test "est différent de" |
| / | division | = | affecte une valeur à une variable |
| ** | élévation à la puissance | print | affiche à l'écran |
| // | quotient dans la division euclidienne | and | opération logique (booléenne) : "et" |
| or | opération logique (booléenne) : "ou" inclusif | not | opération logique (booléenne) : "non" |

2. Chaîne de caractères

Indiquer dans le tableau ci-dessous de quelle instruction/fonction/indication il s'agit :

| | |
|---|--|
| + appliqué à deux données de type <code>str</code> | concatène les deux chaînes de caractères (chaîne de caractère) |
| <code>in</code> | teste si une chaîne de caractère est dans une autre (booléen) |
| <code>len</code> | donne la longueur d'une chaîne de caractère (entier) |
| <code>chaîne[3:5]</code> | retourne la chaîne des 4e et 5e caractères de la chaîne "chaîne" |

Justifier que l'instruction : `7 + 3 + str(2)` renvoie un message d'erreur.

Proposer deux corrections : l'une qui renvoie une chaîne de caractère (à préciser), l'autre qui renvoie un entier (à préciser).

La seconde opération "+" est demandée entre deux types de données incompatibles pour cette opération : un entier (3) et une chaîne de caractère (`str(2) = '2'`).

Correction 1 : `str(7+3) + str(2)` renvoie : `'102'`

Correction 1 : `str(7+3+2)` renvoie : `'12'`

Correction 1 : `str(7) + str(3) + str(2)` renvoie : `'732'`

Correction 2 : `7 + 3 + 2` renvoie 12

Correction 2 : `7 + 3 + int(str(2))` renvoie 12

Correction 2 : `int(str(7+3) + str(2))` renvoie : 102

3. Fonction

Écrire une fonction **puissance** qui prend en entrée deux nombres réels et retourne le premier élevé à la puissance du second :

```
def puissance(a,b):
    return(a**b)
```

autre possibilité :

```
def puissance(a,b):
    p=(a**b)
    return(p)
```

4. Liste

```
x = range(6)
for n in x:
    print(n)
```

0
1
2
3
4
5

Indiquer ce qu'affiche ce programme

5. Instructions conditionnelles

Écrire une fonction **puissance2** qui prend en entrée deux nombres réels et retourne le plus grand élevé à la puissance du plus petit.

```
def puissance2(a,b):
    if a>b:
        return(a**b)
    else:
        return(b**a)
```

autre possibilité :

```
def puissance2(a,b):
    if a>b:
        p=(a**b)
    else:
        p=(b**a)
    return(p)
```

Boucle séquentielle (boucle for)

Écrire une fonction **puissance3** qui prend en entrée un réel et un entier naturel non nul, et qui élève le premier à la puissance du second mais en utilisant uniquement la fonction mathématique multiplier et une boucle séquentielle.

```
def puissance3(x,n):
    p=1
    for k in range(n):
        p=p*x
    return(p)
```

Boucle conditionnelle (boucle while)

Écrire une fonction **puissance4** qui prend en entrée un réel et un entier naturel non nul, et qui élève le premier à la puissance du second mais en utilisant uniquement la fonction mathématique multiplier et une boucle conditionnelle.

```
def puissance4(x,n):
    p=1
    k=0
    while k<n:
        p=p*x
        k+=1
    return(p)
```

Quel serait le code de l'instruction : **i+=1** ? : **i=i+1**