ISM MPSI/PCSI, Informatique
Année 2023/2024

TP Informatique 1

Exercice 1

On rappelle que l'instruction `type` renvoie le type d'un objet. Déterminer le résultat des instructions suivantes puis le vérifier par expérimentation :

1. `1+2, 1-2, 2*3-1, 2**5, 2**(5-1), 2**5-1, (2**2)**3, 2**2**3, abs(-2)`
2. `type(1), type(-1), type(2**2023), type(-2**2023)`
3. `1234%10, 1234//10, (1234//10)%10, 1234//100`
4. `15%2 ,15//2, 2//5, 2%5`

Exercice 2

Déterminer le résultat des instructions suivantes puis le vérifier par expérimentation :

1. `type(1.), type(0.), type(1/1), type(4/2), type(1+0.)`
2. `1.+2, 2.**2, 2**(1/2)`
3. `10**1, 10**-3, 1e1, 1e3, 1e-3`
4. `.1+.2, .1+.7, 49*(1/49), (2**(1/2))**2`

Exercice 3

Déterminer le résultat des instructions suivantes puis le vérifier par expérimentation :

1. `int(1.), int(1.5), int(-1.), int(-.9)`
2. `float(1), float(-1), float(2**1023), float(2**1024)`

Exercice 4

Déterminer le résultat des instructions suivantes puis le vérifier par expérimentation :

1. `type(True), type(False)`
2. `not True, True or False, True and False, not(True and False), not True and False`
3. `1>2, 1>0, 1!=0, 2==1+1, 1>0., 2.==1+1`

Exercice 5

Déterminer le résultat des instructions suivantes puis le vérifier par expérimentation :

1. `int(True), int(False), bool(1), bool(0), bool(2), bool(-3.4)`
2. `0==False, True ==1., 1+True, 1.*False`

Exercice 6

Expérimenter les instructions suivantes et proposer une explication aux comportements observés :

`1/0, False or 1/0==1, True or 1/0==1`

Exercice 7

Saisir l'instruction `import numpy as np` puis expérimenter les instructions suivantes :

1. `np.pi, np.cos(np.pi), np.sin(np.pi)`
2. `np.sqrt(4), np.sqrt(2), np.floor(5.8)`
3. `np.floor(1.3), int(1.3), np.floor(-.8), int(-.8)`

Exercice 8

Expérimenter les instructions suivantes :

1. `a=1, type(a), a=1., type(a)`
2. Saisir `a=10, a+=1` puis afficher le contenu de `a`, saisir `a//=2` puis afficher le contenu de `a`.

Exercice 9

Expérimenter les instructions suivantes :

1. Saisir `a,b=1,2` puis afficher le contenu de `a` et de `b` ;
2. Saisir `a,b=1,2` puis `a,b=b,a` et afficher le contenu de `a` et `b`.

Exercice 10

Dans cet exercice, les saisies seront faites dans l'éditeur : ouvrir un nouveau fichier puis l'enregistrer dans son répertoire personnel (si possible) sous un nom adapté, `TP01_10.py` par exemple). Une fois les saisies terminées, exécuter le programme (onglet `Exécuter` puis `Démarrer le scipt` ou `Ctrl+Shift+E`).

1. Saisir `a=123` puis `print(a)` et exécuter le programme.
2. Saisir `import numpy as np` puis `print("cos(pi)=",np.cos(np.pi)," sin(pi)=",np.sin(np.pi))` et exécuter le programme.

Exercice 11

Dans cet exercice, les saisies seront faites dans l'éditeur. Un nombre à p chiffres est dit *narcissique* s'il est égal à la somme de ses chiffres à la puissance p. Par exemple, le nombre 153 est narcissique :

$$1^3 + 5^3 + 3^3 = 153$$

1. Saisir `a=...` en remplaçant les points par un nombre à 3 chiffres de votre choix.
2. Programmer l'affichage de `True` si le nombre `a` est narcissique et de `False` sinon. Le programme doit fonctionner quel que soit le choix du nombre `a` à 3 chiffres.

Exercice 12

Dans cet exercice, les saisies seront faites dans l'éditeur. Un nombre est dit *palindrome* si son écriture décimale est symétrique. Par exemple, les nombres 121 ou 123321 sont des nombres palindromes.

1. Saisir `a=...` en remplaçant les points par un nombre à 4 chiffres de votre choix.
2. Programmer l'affichage de `True` si le nombre `a` est palindrome et de `False` sinon. Le programme doit fonctionner quel que soit le choix du nombre `a` à 4 chiffres.