

1 Les bases de python3

Les types

Les différents types entiers, flottants, complexes, chaînes de caractères, booléens, listes, tuples et dictionnaires.

- *entier* : 3
- *flotant* : 2.3
- *complexe* : 1+2j
- *chaînes de caractères (string)* : 'ISN'
- *Boléen* : True
- *Listes* : [0,1,2,3]
- *Tuples* : (0,1,2,3)
- *Dictionnaires* : 'zero' :0, 'un' :1, 'deux' :2, 'trois' :3
- *Byte* : b'toto'
- On aura toujours à l'esprit que $0.2 + 0.7 = 0.8999999999999999 \neq 0.9$
- *int()* permet de convertir, un nombre ou une chaîne de caractère en un entier.
- *float()* permet de convertir, un nombre ou une chaîne de caractère en un flottant.
- On utilise le caractère # pour écrire un commentaire

Les opérateurs

| Opération | algorithmique | python |
|-------------------------------------|-----------------------|---------|
| Addition | 2+3 | 2+3 |
| Soustraction | 12-5 | 12-5 |
| Multiplication | 3*6 | 3*6 |
| Division | 7/2 | 7/2 |
| Quotient de la division euclidienne | 7 div 2 ou div(7,2) | 7//2 |
| Reste de la division euclidienne | 7 mod 2 ou mod(7,2) | 7%2 |
| puissance | 7 ² | 7**2 |
| racine carrée | $\sqrt{2}$ ou sqrt(2) | sqrt(2) |

Variables

- Python utilise le symbole = pour affecter une valeur à une variable.
- Attention à ne pas confondre $a = 12$ et $a == 12$.

'Python3'

```
A=3; B=5
C='toto'
```

Entrées

- *input("texte")* permet de saisir du texte pour un programme.
- Il faudra éventuellement convertir ce texte dans le type voulu avec *int()* ou *float()*

```
a=input("Entrer un texte a: ")
b=int(input("Entrer un entier b: "))
c=float(input("Entrer un réel c: "))
```

Affichage

- On utilise de préférence la méthode *format()* pour afficher du texte et des variables.

```
print("La valeur de la variable a est",str(a),".") # si a est un nombre
print("La valeur de la variable a est " + str(a) + ".") # autre forme
print("La valeur de la variable a est {}".format(a)) # je préfère
print("Le produit de {} par {} est {}".format(a,b,a*b))
print("{}*{}={2} et {0}/{1}={3}".format(a,b,a*b,a/b))
```

Connecteurs logiques

| algorithmique | python |
|---------------|------------------|
| $a = b$ | $a == b$ |
| $a \neq b$ | $a != b$ |
| A et B | A and B A & B |

| algorithmique | python |
|---------------|-----------------|
| A ou B | A or B A B |
| A xor B | $A \wedge B$ |
| non A | not(A) |

Condition SI

```
if x<=0:
    print('Pas de racines')
```

```
if x<=0:
    print('Pas de racines')
else:
    print('1 ou 2 racines')
```

```
if x<=0:
    print('Pas de racines')
elif x==0:
    print('1 racines double')
else:
    print('2 racines distinctes')
```

```
print("Saisissez deux valeurs numériques")
a=float(input("Saisir a: "))
b=float(input("Saisir b: "))
if a==b :
    print("Vous avez saisi deux fois la même valeur, à savoir {}".format(a))
else :
    print("Vous avez saisi deux valeurs différentes {} et {}".format(a,b))
```

Boucle pour

```
for i in range(4,21,2): # 4 6 ... 20
    print(i)
for i in range(6):      # de 0 à 5 sinon utiliser range(1,7) pour 1 2 ...6
    print(i)
```

La syntaxe générale est *for i in range(m,n,p)* :
i prend alors toutes les valeurs de *m* à *n-1* par pas de *p*

Tant que

```
i=1
while i<=5:
    print(i)
    i=i+1 #où en plus concis i+=1
#À la sortie de la boucle i=6
```