

Programmation Python - Algorithme - Fiche de cours

1. Entrées, sorties et variables

Pour lire un message on peut utiliser l'instruction :

```
variable = input(« Message »)
```

Pour afficher un message on peut utiliser l'instruction :

```
print (« Message », variable)
```

Pour convertir le type des variables on peut utiliser :

str() pour les chaînes de caractères

float() pour les nombres décimaux

int() pour les nombres entiers

2. Calcul et test avec Python

Les opérations de base :

+	addition	%	reste de la division
-	soustraction	==	est égal ?
*	multiplication	!=	est différent ?
/	division	<	est inférieur ?
**	puissance	>	est supérieur ?
//	division entière	<=	est inférieur ?
=	Affecter une valeur	>=	est inférieur ?

3. Les listes de valeur

```
liste = [a,b,c]
```

```
liste[n]
```

```
liste.append(a)
```

```
len(liste)
```

```
min(liste)
```

```
max(liste)
```

```
sum(liste)
```

```
liste.sort()
```

Crée une variable *liste* avec les valeurs a,b,c

Renvoie la nième valeur de la variable *liste*

Ajoute a en fin de la variable *liste*

Renvoie le nombre de valeurs de la variable *liste*

Renvoie le minimum de la variable *liste*

Renvoie le maximum de la variable *liste*

Renvoie la somme des valeurs de la variable *liste*

Trie les valeurs par ordre croissant de la variable *liste*

4. if else

```
if condition :
```

```
    Instruction 1
```

```
    Instruction 2
```

```
else :
```

```
    Instruction 1
```

```
    Instruction 2
```

5. for

```
For n in range (nombre) :    Execute nombre-1 séquences
```

```
    Instruction 1
```

```
    Instruction 2
```

```
    ...
```

6. while

```
while condition :
```

```
    Instruction 1
```

```
    Instruction 2
```

```
    ...
```

7. Importer des fonctions

`from fichier import *`

Les divers fichiers (bibliothèques) en Python :

- *math* : contient une partie des fonctions mathématiques
- *cmath* : contient les fonctions mathématiques pour nombres complexes
- *fractions* : contient les fonctions mathématiques pour nombres rationnels
- *random* : contient les fonctions de tirages aléatoires et les fonctions de probabilités

8. Définir une fonction

```
def nom_fonction (paramètre1, paramètre2, etc...):  
    instructions  
    ...  
    return résultat
```

9. Fonctions mathématiques : « *math* »

fabs(x)	: retourne la valeur absolue de x
factorial(x)	: retourne la valeur absolue de x
exp(x)	: retourne l'exponentielle de x
log(x)	: retourne le logarithme naturel de x
log10(x)	: retourne le logarithme décimal de x
pow(x,y)	: retourne x puissance y
sqrt(x)	: retourne la racine carrée de x
cos(x)	: retourne la valeur de cos(x)
sin(x)	: retourne la valeur de sin(x)
tan(x)	: retourne la valeur de tan(x)

acos(x)	: retourne la valeur de arccos(x)
asin(x)	: retourne la valeur de arcsin(x)
atan(x)	: retourne la valeur de arctan(x)
pi	: retourne la valeur du nombre pi
e	: retourne la valeur du nombre e

10. Fonctions mathématiques : « *fractions* »

Fraction(a,b)	: définit une fraction irréductible = a/b
Fraction(nombre)	: définit la fraction irréductible = a/b
fraction.numerator	: retourne le numérateur de <i>fraction</i>
fraction.denominator	: retourne le dénominateur de <i>fraction</i>
gcd(a,b)	: retourne le PGCD(a,b)

11. Fonctions mathématiques : « *random* »

random(x)	: retourne un nombre décimal entre 0,0 et 1,0
randint(1,5)	: retourne un nombre entier entre 1 et 5
uniform(a,b)	: retourne un nombre décimal entre a et b
expovariate(a)	: retourne un nombre décimal distribué selon la loi exponentielle de paramètre a
normalvariate(mu,sigma)	: retourne un nombre décimal distribué selon la loi normale de paramètre mu et sigma

12. Logiciel à télécharger

PC : <https://edupython.tuxfamily.org/>

Smartphone : « QPython 3 » pour Android ou Iphone