

Entrée / sortie

Exercice 1

Demander à la personne son nom, puis son prénom, puis son parfum de glace préféré et afficher une phrase sur le modèle " Bonjour Julien Kergot. Tu aimes les glaces parfum chocolat " qui utilisera les informations récoltées.

```
nom = input("Quel est ton nom ? ")
prenom = input("Quel est ton prenom ? ")
gout = input("Quel est ton parfum de glace préféré ? ")
print("bonjour "+nom+ " "+prenom+" tu aimes la glace au/à la "+g
```

Exercice 2

- 1) Donner le discriminant d'une fonction trinôme dont les trois coefficients ont été donné par l'utilisateur
- 2) Donner les solutions si elles existent

```
from math import *

print("f(x)=ax²+bx+c")
a = int(input("a=? ")) # input me donne une chaine de caractère
b = int(input("b=? ")) # mais moi j'ai besoin d'un nombre
c = int(input("c=? ")) # j'utilise donc int() qui transforme les
delta = b*b-4*a*c # chaines en nombres
print("le discriminant du polynome est : ",delta)
#ici on ne peut écrire "..."+delta, on ne peut utiliser + qu'entre
#deux objets du même type
if delta>=0 :
    sol1=(-b-sqrt(delta))/(2*a)
    sol2=(-b+sqrt(delta))/(2*a)
    print("les solutions sont :")
    print("x1=",sol1)
    print("x2=",sol2)
else :
    print("il n'y a pas de solution réelles")
```

Test

Exercice 3

Demander à l'utilisateur un nombre et dire si celui-ci est un multiple de 7

```
a = int(input("donner un nombre : "))

if a%7==0 :
    print(a, " est divisible par 7")
else :
    print(a, "n'est pas divisible par 7")
```

Exercice 4 : division euclidienne

Demander à l'utilisateur le dividende et le diviseur, lui donner en retour le quotient et le reste.

```
print("division euclidienne")
dividende = int(input("donner le dividende : "))
diviseur = int(input("donner le diviseur : "))
reste = dividende%diviseur
quotient = (dividende-reste)/diviseur
print("le quotient vaut : ",quotient)
print("le reste vaut : ",reste)
print(dividende,"=",quotient,"*",diviseur,"+",reste)
```

Exercice 5

Demander à l'utilisateur la note qu'il a eu, puis si celle-ci est supérieure à 16 vous afficherez que ça correspond à un A, sinon si celle-ci est supérieure à 12 sa note correspond à un B ainsi de suite.

```
note = float(input("donner votre note : "))
if note>16 :
    noteL="A"
elif note>12 :
    noteL="B"
elif note>8 :
    noteL="C"
elif noteL>4 :
    noteL="D"
elif note>0 :
    noteL="E"
else :
    noteL="F"
print("la note en format lettres sera : ",noteL)
```

Exercice 6

Demander à une personne son année de naissance, et suivant celle-ci vous afficherez « tu es majeur », « tu n'es pas majeur » ou vous demanderez le numéro

du mois de naissance, et de là de nouveau vous afficherez si elle est majeure ou pas ou vous demanderez le jour et vous donnerez votre verdict final.

```
#Amois = float(input("Quel mois est on ? "))
#Ajour = float(input("Quel jour est on ? "))
#Année = float(input("Quel année est on ? "))

Année,Amois,Ajour = 2017,9,20
# les trois valeurs sont associées aux variables respectives

Pannée = float(input("Quel année es tu né ? "))
if Pannée>=Année-17 :
    print("tu es mineur")
elif Pannée<=Année-19 :
    print("tu es majeur")
else :
    Pmois = float(input("Quel mois es tu né ? "))
    if Pmois>Amois :
        print("tu es mineur")
    elif Pmois<Amois:
        print("tu es majeur")
    else :
        Pjour = float(input("Quel jour es tu né ? "))
        if Pjour>Ajour :
            print("tu es mineur")
        elif Pjour<Ajour :
            print("tu es majeur")
        else :
            print("joyeux 18ème anniversaire")
```

Exercice 7

Ecrire un programme qui simule le lancer d'un dé à 20 faces tant que le résultat obtenu n'est pas égal à 10. Le programme doit afficher au bout de combien d'essai il a obtenu le résultat 10

```
from random import *

lancé,nbrjet = 0,0
while lancé!=10 :
    nbrjet+=1
    lancé = randint(1,20)
print("il a fallu ",nbrjet," lancés pour obtenir un dix")
```

Boucles

Exercice 8

Facile : Créer une fonction qui pourra déterminer la somme de tous les entiers compris entre deux entiers donnés.

```
somme = 0
a=int(input("valeur de départ : "))
b= int(input("valeur finale : "))
for i in range(a,b+1) :
    somme+=i
print ("la somme vaut : ",somme)
```

Moyen : faire la somme des nombres de la suite de Fibonacci inférieurs à un nombre donné.

```
premier, second, nouveau = 1,2,3
somme = premier+second
borne =int(input("valeur à ne pas dépasser : "))
while nouveau<=borne :
    somme+=nouveau
    premier, second, nouveau = second, nouveau, second + nouveau
    print(nouveau)
print ("la somme vaut : ",somme)
```

Moyen+ : faire la somme des nombres premiers inférieurs à un nombre donné

```
borne = int(input("borne à ne pas dépasser : "))
somme = 0
for i in range(1,borne+1):
    ajouter = True
    for j in range(2,i) :
        if i%j==0 :
            ajouter = False
    if ajouter :
        print(i)
        somme+=i

print ("la somme vaut : ",somme)
```

Exercice 9

créer une fonction qui a tout x associe le nombre $x^3 - 3x + 1$

Afficher les images des nombres entiers entre -10 et 10 , par exemple : f(0)=1

```
def f(x) :
    return x*x*x-3*x+1

for i in range(-10,11) :
    print("f(",i,")=",f(i))
```

Exercice 10

Créer un programme demandant un nombre et donnant tous les diviseurs de ce nombre.

```
nombre=int(input("nombre à étudier : "))
print("diviseurs de ",nombre," : ")
for i in range(1,nombre+1) :
    if nombre%i==0 :
        print(i)
```

Exercice 11

Créer une fonction pour déterminer la factorielle d'un nombre.

```
def factorielle(n):
    facto = 1
    for i in range(1,n+1) :
        facto*=i
    return facto
```

```
nombre=int(input("nombre dont on veut la factorielle : "))
print("la factorielle de ",nombre," vaut : ",factorielle(nombre))
```

Version pro : faire le programme de manière récursive.

```
def factorielle(n):
    if n == 1 :
        return 1
    else :
        return n*factorielle(n-1)
```

Exercice 12

Ecrire un programme qui demande à l'utilisateur d'entrer un nombre n et qui affiche à l'écran la table de multiplication de ce nombre. Par exemple si ce nombre est 5 on aura :

5 x 1 = 5 puis à la ligne suivante : 5 x 2 = 10 et ainsi de suite

```
nombre=int(input("nombre dont vous voulez la table : "))
for i in range(1,11) :
    print(i,"*",nombre,"=",i*nombre)
```

Exercice 13

Ecrire un programme qui demande à l'utilisateur d'entrer un nombre n compris entre 1 et 10 et qui afficherait par exemple pour n= 3 :

```
*           aide :
**          pour ne pas revenir à la ligne après une instruction print,
***         on utilise : print("o",end="")
```

```
for i in range(1,11) :
    for j in range(1,i+1) :
        print("*",end="")
    print("")
```

Exercice 14

Sans vous servir de la fonction intégrée dans python, écrire un programme demandant un nombre en base 10 et le convertissant en base 2

```
nombre=int(input("nombre à convertir : "))
result=""
while nombre!=0 :
    #print(nombre%2)
    result = str(int(nombre%2))+result
    nombre = (nombre-nombre%2)/2
print (result)
```

Version pro : Faire la même chose en base hexadécimale

```
nombre=int(input("nombre à convertir : "))
result=""
```

```
while nombre!=0 :
    #print(nombre%2)
    interm = nombre%16
    if interm<=9 :
        result = str(int(nombre%16))+result
    elif interm==10:
        result = "a"+result
    elif interm==11:
        result = "b"+result
    elif interm==12:
        result = "c"+result
    elif interm==13:
        result = "d"+result
    elif interm==14:
        result = "e"+result
    elif interm==15:
        result = "f"+result
    nombre = (nombre-nombre%16)/16
print (result)
```

Exercice 15

Ecrire un programme qui affichera un titre à l'intérieur d'un cadre fait d'étoiles.

Il demande un titre et un nombre de caractère d'écart verticalement puis horizontalement.

Par exemple avec " coucou les gens " 1 et 3 il affichera :

```
*****
*           *      Indication : la commande len(chaine) donne
* coucou les gens *      le nombre de caractère dans la chaine
*           *
*****
```

```
def linestar(long) :
    for i in range(long) :
        print("*",end="")
    print("")

def lineempty(long):
    print("*",end="")
    for i in range(1,long-1) :
        print(" ",end="")
    print("")

def linetitle(paddh,titre) :
    print("*",end="")
    for i in range(paddh) :
        print(" ",end="")
    print(titre,end="")
    for i in range(paddh) :
        print(" ",end="")
    print("")
```

```
texte = input("quel est le titre ? ")
padv = input("quel est le remboursement vertical ? ")
padh = input("quel est le remboursement horizontal ? ")
#texte,padv,padh = "titre à encadrer",3,7
```

```
longueur = len(texte)+2*padh+2
linestar(longueur)
for i in range(padv) :
    lineempty(longueur)
linetitle(padh,texte)
for i in range(padv) :
    lineempty(longueur)
linestar(longueur)
```

Travail sur les chaînes de caractères

Exercice 16

Créer une fonction pour écrire une chaîne de caractère à l'envers

```
texte = input("quel est le titre ? ")
textinv = ""
for i in range(len(texte)):
    textinv = texte[i]+textinv
print(textinv)
```

Exercice 17

Créer une fonction pour déterminer si un mot est un palindrome

On rajoutera au programme précédent les lignes suivantes :

```
if texte==textinv :
    print("c'est un palindrome")
else :
    print("ce n'est pas un palindrome")
```

Exercice 18

Créer une fonction qui déterminera dans une chaîne de caractère quel est le mot le plus long

```
texte = input("quel est le texte ? ")
chaine = texte.split(" ")
longueurs = []
maxlong = 0
for i in range (len(chaine)) :
    longueurs = longueurs + [len(chaine[i])]
    if len(chaine[i])>maxlong :
        maxlong = len(chaine[i])
        compteur = 1
    elif len(chaine[i])==maxlong :
        compteur += 1
print("Il y a ",compteur," mot(s) le(s) plus long(s).")
print("Ils font ",maxlong," caractères et sont :")
for i in range (len(longueurs)) :
    if longueurs[i] == maxlong :
        print(chaine[i])
```

Exercice 19

Créer une fonction qui va mettre en majuscule les premiers mots de chacune des phrases d'une chaîne de caractère.

Indication : jouez à utiliser des instructions `lower()`, `upper()`, `capitalize()` et `title()` sur des chaînes de caractère.

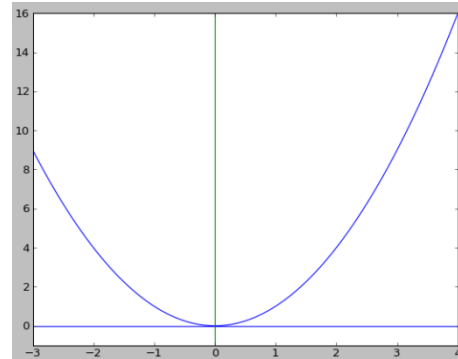
Matplotlib/PyLab

PyLab est une bibliothèque qui sert à la représentation graphique des données

Fonctions

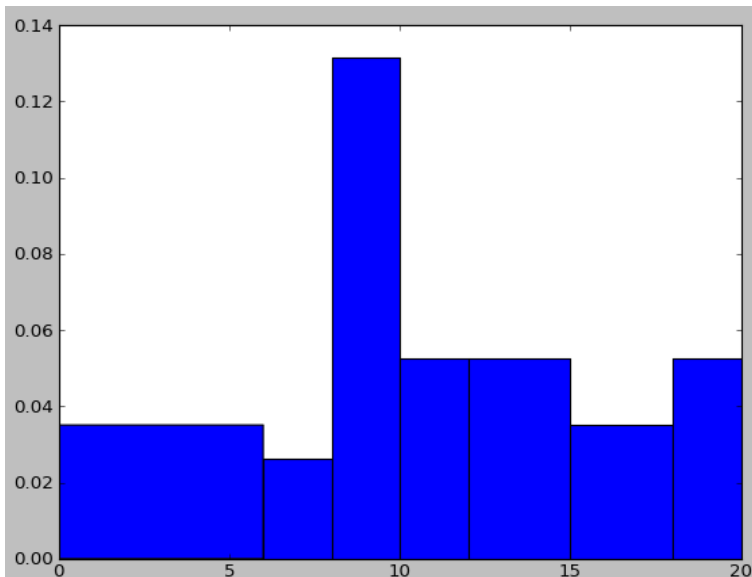
```
from pylab import *  
  
x=arange(-3,4,0.01)  
plot(x,x**2)  
axis([-3,4,-1,16])  
axhline(color="blue")  
axvline(color="green")  
show()
```

arange est une variante de range qui me permet d'avoir les nombres compris entre les bornes -3 et 4 en allant de 0,01 en 0,01



Histogramme :

```
from pylab import *  
valeurs=[2,5,5,5,7,8,8,9,9,9,10,10, 12,13,13,15,17,18,19]  
bornes=[0,6,8,10,12,15,18,20]  
hist(valeurs,bornes,normed=True)  
show()
```



Suites

Petit programme pour représenter les 20 premiers termes de la suite

$$\begin{cases} u_0 = 50 \\ u_{n+1} = -0,5 U_n + 30 \end{cases}$$

```
from pylab import *  
  
nbrTermes = 20  
X=[0] # pour le rang 0  
Y=[50] # U(n) vaudra 50  
minimum, maximum = Y[0],Y[0]  
for i in range(1,nbrTermes):  
    newval=Y[i-1]*(-0.5)+30 # calcul du terme suivant de la suite  
    if minimum>newval :  
        minimum = newval #ajustement du Ymin  
    if maximum<newval :  
        maximum = newval #ajustement du Ymax  
    X=X+[i] #on crée les coordonnées du point  
    Y=Y+[newval] #suivant  
  
amplitude=maximum - minimum  
minimum, maximum = minimum-0.1*amplitude,maximum+0.1*amplitude
```

```
scatter(X,Y,color="red")  
axis([-1,nbrTermes,minimum, maximum]) #on communique à l'ordinateur  
# les bornes du repère  
axhline(color="blue")  
axvline(color="green")  
show()
```

