

## 05. Boucle *while*

Nous utiliserons dans ce TP le langage de programmation *Python* au moyen de l'environnement de développement *IDLE*.

**Exemple 1.** Affichage des puissances de 2 inférieures à un entier  $n$  : algorithme et programme en Python

**Entrée:** variable entière  $n$   
**Sortie:** affichage des puissances de 2 inférieures à  $n$   
**Début**  
 Lire  $n$   
 $p \leftarrow 1$   
**TantQue**  $p < n$  **faire**  
     **Afficher**  $p$   
      $p \leftarrow 2 * p$   
**FinTantQue**  
**Fin**

```
#Entrée : n entier
#Sortie : affichage des puissances de 2 inférieures à n
n=int(input("valeur de n?"))
p=1
while p<n:
    print(p)
    p=2*p
```

**Exercice 1.** Enregistrer ce programme dans un fichier nommé *puissancesdedeux.py*, puis l'exécuter et le tester. Créer puis tester un programme permettant d'obtenir la plus petite puissance de deux supérieure ou égale à un entier  $n$  quelconque.

**Exercice 2.** Créer puis tester un programme permettant d'afficher la liste des multiples de sept inférieurs ou égaux à un entier  $n$  quelconque.

**Exercice 3.** Créer puis tester un programme permettant d'afficher la liste des carrés inférieurs ou égaux à un entier  $n$  quelconque.

**Exercice 4.** Créer puis tester un programme permettant de calculer le nombre d'années nécessaire pour doubler un capital placé à intérêts composés avec un taux annuel de  $t\%$ .

**Exercice 5.** Créer puis tester un programme permettant de déterminer combien de fois un entier  $n$  non nul donné est divisible par 2.

**Exercice 6.** Créer puis tester un programme permettant de calculer le plus grand diviseur commun de deux entiers  $m$  et  $n$  au moyen de soustractions successives.

## Réponses

- 1) #Entrée : n entier  
 #Sortie : affichage de la plus petite puissance de deux supérieure ou égale à n  

```
n=int(input("valeur de n?"))
p=1
while p<n:
    p=2*p
print("la plus petite puissance de deux supérieure ou égale à",n,"est",p)
```
- 2) #Entrée : n entier  
 #Sortie : affichage de la liste des multiples de sept inférieurs ou égaux à n  

```
n=int(input("valeur de n?"))
m=0
while m<=n:
    print(m)
    m=m+7
```
- 3) #Entrée : n entier  
 #Sortie : affichage des carrés inférieurs ou égaux à n  

```
n=int(input("valeur de n?"))
k=0
while k**2<=n:
    print(k**2)
    k=k+1
```
- 4) #Entrée : t flottant  
 #Sortie : affichage du nombre d'années nécessaire pour doubler un capital placé au taux de t%  

```
t=float(input("valeur de t?"))
c=1
i=0
while c<2:
    c=c*c*t/100
    i=i+1
print("le nombre d'années nécessaire pour doubler un capital placé au taux de",t,"%","est",i)
```
- 5) #Entrée : n entier  
 #Sortie : affichage du nombre de fois que l'on peut diviser n par 2  

```
n=int(input("valeur de n?"))
k=n
i=0
while k%2==0:
    k=k//2
    i=i+1
print("le nombre de fois que l'on peut diviser",n,"par",2,"est",i)
```
- 6) #Entrée : m et n entiers  
 #Sortie : affichage du plus grand diviseur commun de m et n  

```
m=int(input("valeur de m?"))
n=int(input("valeur de n?"))
a=m
b=n
while a!=b:
    if a<b:
        b=b-a
    else:
        a=a-b
print("le plus grand diviseur commun de",m,"et",n,"est",a)
```