

TD : Structure conditionnelle if/else

1. a) Rappeler la signification de la commande `np.floor` accessible depuis la bibliothèque `numpy`.

b) Compléter et expliquer le fonctionnement du programme suivant ?

```

7  n=int(input('rentrer un entier n'))
8  if np.floor(n/2)==n/2 :
9      print('n est pair')
10 else:
11     print('n est ...')
```

c) Remplacer la ligne 8 par une condition utilisant les puissances de (-1).

2. Écrire un programme qui demande 3 réels a , b et c et qui affiche si le trinôme $ax^2 + bx + c$ possède des racines ou non.

3. a) Compléter le programme suivant pour qu'il affiche le nombre de racines du trinôme $ax^2 + bx + c$.

```

7  a=float(input('rentrer un réel a non nul'))
8  b=float(input('rentrer un réel b'))
9  c=float(input('rentrer un réel c'))
10 if b**2-4*a*c....:
11     print('le trinôme ne possède pas de racine')
12 elif b**2-4*a*....:
13     print('le trinôme a une unique racine')
14 else:
15     print('....')
```

4. Écrire un programme qui demande une température T qui affiche : "gelé" si $T < 0$, « frisquet » si $0 \leq T < 10$, « tiède » si $10 \leq T < 20$, « chaud » si $20 \leq T < 30$, « brûlant » si $T \geq 30$.

5. Écrire un programme pour chaque fonction qui demande à l'utilisateur un réel x et qui affiche la valeur de $f(x)$ dans le cas où :

a)
$$f(x) = \begin{cases} \frac{1-e^{-x}}{x} & \text{si } x \neq 0 \\ 1 & \text{si } x = 0 \end{cases}$$

b)
$$f(x) = \begin{cases} |x| & \text{si } x \leq 0 \\ \frac{1}{x} e^{\frac{-1}{x}} & \text{si } x > 0 \end{cases}$$

$$c) \quad f(x) = \begin{cases} \ln(1+x) & \text{si } x > 0 \\ \frac{x+x^2}{1-x} & \text{si } x \in]-1, 0] \\ 0 & \text{si } x \leq -1 \end{cases}$$

6. Soit la suite (u_n) définie par $u_0 = k$ et $\forall n \in \mathbb{N}$, si u_n pair, $u_{n+1} = \frac{u_n}{2}$, si u_n impair, $u_{n+1} = 3u_n + 1$.

- Calculer à la main les 5 premiers termes de la suite lorsque $k=10$ et $k=5$.
- Compléter le programme suivant pour qu'il affiche le n ième terme de la suite lorsque l'utilisateur rentre des entiers k et n .

```

7  n=int(input('retrrer un entier n'))
8  k=int(input('rentrer le premier terme k'))
9  u=...
10 for k in range(1,...):
11     if np.floor(u/2)==u/2:
12         ...
13     else:
14         ...
15 print(u)
```

Corrigé

1.

- a) Elle calcule la partie entière de x , c'est-à-dire le plus grand entier inférieur ou égal à x .
- b) Le programme affiche la parité de n , en effet partie entière de $n/2$ est égal à $n/2$ si et seulement si $n/2$ est entier si et seulement si n est entier
- c) `if (-1)**n==1. # car cela fait 1 uniquement lorsque la puissance est paire`

2.

```
a=float(int('rentrer la valeur de a, réel non nul'))
b=float(int('rentrer la valeur de b'))
c=float(int('rentrer la valeur de c'))
d=b**2-4*a*c      # on calcule le discriminant
if d>=0 :
    print('le trinôme a des racines')
else :
    print('le trinôme n'a pas de racines')
```

3.

- a) On complète dans l'ordre les ... :
`<0`
`==0`
`'le trinôme a deux racines'`
- b) Il suffit de mettre ligne 13 : `print(-b/(2*a))`
 Et ligne 15 `print((-b+sqrt(b**2-4*a*c))/(2*a) , (-b-sqrt(b**2-4*a*c))/(2*a))`

4.

```
T=float(input('rentrer la température'))
if T<0 :
    print('gelé')
elif T<10 :
    print('frisquet')
elif T<20 :
    print('tiède')
elif T<30 :
    print('chaud')
else :
    print('brulant')
```

5.

- a) `x=float(input('rentrer la valeur du réel x'))`
`if x !=0 :`

```

        print((1-exp(-x))/x)
    else :
        print(1)

b) x=float(input('rentrez la valeur du réel x'))
    if x>=0 :
        print(floor(x))
    else :
        print(1/x*exp(-1/x))

c) x=float(input('rentrez la valeur du réel x'))
    if x>0 :
        print(log(1+x))
    elif x<=0 :
        print((x+x**2)/(1-x))
    else :
        print(0)

```

6

```

b) n=int(input('rentrez la valeur de l'entier naturel n'))
    k=int(input('rentrez la valeur de k'))
    u=k
    for i in range(0,n):
        if floor(u/2)==u/2:          # vérifie si u est pair
            u=u/2
        else:
            u=3*u+1

```