

Devoir surveillé n°4 – NSI  
15/12/2020**Exercice 1 (2 points)**

1. (a) Quel est le nombre de sommes effectuées, en fonction de  $n$ , dans l'algorithme ci-contre ? Expliquer.
- (b) En déduire la complexité (linéaire, quadratique ou autre) de l'algorithme.

```
x=1
for i in range(n):
    for j in range(1,5):
        print(x+j)
    x+=1
```

**Exercice 2 (5 points)**

1. Réécrire et compléter la fonction `existe` ci-dessous, qui prend en argument une variable  $x$  et une liste  $l$ , qui renvoie `True` si  $x$  est dans  $l$ , et qui renvoie `False` sinon.

```
def existe(x,l):
    for e in l:
        if .....:
            return .....
    return .....
```

2. Écrire une fonction `effaceDoublons` qui prend en argument une variable  $l$  de type list et qui retourne une liste contenant les éléments de  $l$  mais sans doublon.  
Par exemple, `effaceDoublons([2,4,5,2,1,4,7])` retourne la liste `[2,4,5,1,7]`.  
On pourra utiliser la fonction précédente.
3. Quel est le coût de la fonction `effaceDoublons` (linéaire, quadratique ou autre) en fonction de la longueur de la liste donnée en argument ? Expliquer.

**Exercice 3 (9 points)**

On considère ci-contre une fonction écrite en langage Python.  
La variable  $n$  est un entier naturel et  $b$  est un flottant.  
On rappelle que :

- $n\%d$  renvoie le reste de la division de  $n$  par  $d$  ;
- $n//d$  renvoie le quotient de la division de  $n$  par  $d$ .

1. Que retourne la fonction lorsque  $b=2$  et  $n=4$  ?  
Détaillez l'exécution de l'algorithme.
2. Démontrer que la boucle se termine.
3. Démontrer que la propriété «  $z \times x^k = b^n$  et  $k \geq 0$  » est un invariant de la boucle.
4. En déduire la valeur retournée, en fonction de  $b$  et  $n$ .

```
def fonction(b,n):
    x = b
    k = n
    z = 1
    while k != 0:
        if k%2 == 1:
            z = z*x
        x = x*x
        k = k//2
    return z
```