

DS 1

le 26/09/2026 – Durée : 2h

- L'usage de la calculatrice n'est pas autorisé.
- La clarté et la précision des raisonnements interviendront pour une grande part dans la notation. La présentation (en particulier les indentations) est également essentielle.
- Le résultat d'une question peut être admis afin de traiter une question suivante ; une fonction peut être utilisée dans la suite d'un exercice même si elle n'a pas été écrite.
- Le barème tiendra compte de la longueur du devoir.

Exercice 1

1.

```
def factorielle(n):
    if n==0:
        return 1
    res=1
    for i in range(1,n+1):
        res=res*i
    return res
```

2.

```
def seuil(M):
    n=0
    f=1 # vaut n!
    while f<=M:
        n+=1
        f=f*n
    return n
```

Le programme à éviter était le suivant :

```
def seuil(M):
    n=0
    while factorielle(n)<=M:
        n+=1
    return n
```

3.

```
def est_divisible(n):
    a=factorielle(n)
    if a%(n+1)==0:
        return True
    else :
        return False
```

4. (a) Elle renvoie la somme $1! + 2! + 3! + 4!$, c'est à dire 33.

(b) Pour chaque calcul de $k!$, $1 \leq k \leq n$, elle effectue $k - 1$ multiplications ; et donc en tout $\frac{n(n-1)}{2}$ multiplications (complexité quadratique).

(c)

```
def mystere2(n):
    s=0
    f=1 # prendra la valeur k!
    for k in range(1,n+1):
        f=f*k
        s=s+f
    return s
```

5. (a)

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

Notons $C(n)$ le nombre d'opérations nécessaires au calcul de `mystere_rec(n)`. Comme `factorielle(n)` nécessite $n - 1$ multiplications, la relation de récurrence vérifiée par $C(n)$ est :

$$C(n) = C(n - 1) + n .$$

Cela conduit à une complexité quadratique.

(b)

```
def mystere_rec2(n):  
    if n==1:  
        return [1,1]  
    else :  
        l=mystere_rec2(n-1)  
        fact=l[0]*n  
        somme=l[1]+fact  
        return [fact,somme]
```

Notons $C_2(n)$ le nombre d'opérations nécessaires au calcul de `mystere_rec2(n)`. La relation de récurrence vérifiée par $C_2(n)$ est (on ne compte que les additions et multiplications) :

$$C_2(n) = C_2(n - 1) + 2 .$$

Cela conduit à une complexité linéaire.

Problème

Vous traiterez le problème suivant à l'exception des questions 5 à 9 (incluses).

Remarque : les parties de ce problème sont largement indépendantes.