

Informatique : devoir n°1 (non surveillé)

Vous pouvez rendre le DM sur papier ou m'envoyer vos programmes par mail. Dans ce cas tous les programmes doivent être écrits dans un unique fichier nommé `DM1_votrenom.py`. Les noms des variables doivent être bien choisis. On peut utiliser `#` pour mettre des commentaires. Les DM d'informatique n'étant pas notés, l'utilisation de ChatGPT ne vous ramènera strictement rien.

Exercice 1

1) Écrire une fonction `aire_disque` qui, recevant un nombre positif r , renvoie l'aire d'un disque de rayon r . On prendra 3,14 comme valeur approchée de π .

```
>>> aire_disque(5)
78.5
```

2) Écrire une fonction `aire_rectangle` qui, recevant deux nombres positifs a et b , renvoie l'aire d'un rectangle de longueur a et de largeur b .

```
>>> aire_rectangle(5, 8)
40
```

Exercice 2 - Coefficients binomiaux

Écrire une fonction `coefficient_binomial` qui, recevant deux entiers n et p , renvoie le coefficient binomial $\binom{n}{p}$. On rappelle que ce coefficient est nul lorsque $p < 0$ ou que $p > n$.

```
>>> coefficient_binomial(15, 4)
1365    # et non 1365.0
>>> coefficient_binomial(15, -4)
0
>>> coefficient_binomial(15, 40)
0
```

Exercice 3 - Nombres premiers

On rappelle que un entier naturel est premier s'il a exactement deux diviseurs dans $\mathbb{N}$: 1 et lui-même. Ainsi 2, 3 et 5 sont premiers, alors que 4, qui a trois diviseurs, ne l'est pas. Remarquons que 1 n'a qu'un diviseur, donc il n'est pas premier.

On considère la fonction suivante, qui, recevant un entier naturel n , renvoie `True` si n est premier et `False` sinon. On pourra utiliser cette fonction dans tout l'exercice.

```
def est_premier(n):
    if n < 2:
        return False
    for k in range(2, n):
        if n % k == 0:
            return False
    return True
```

1) Décrire avec précision quels calculs sont effectués par cette fonction lorsqu'on exécute l'instruction `est_premier(5)` puis lorsqu'on exécute l'instruction `est_premier(6)`.

2) a) Soient n, p, q des entiers naturels. Montrer que si $n = pq$, alors $p \leq \sqrt{n}$ ou $q \leq \sqrt{n}$ (raisonner par contraposée).

b) En déduire une amélioration possible de la fonction.

3) Écrire une fonction `nombre_de_premiers` qui, recevant un entier naturel n , renvoie le nombre de nombres premiers inférieurs ou égaux à n . Par exemple, `nombre_de_premiers(10)` doit renvoyer 4.

4) Écrire une fonction `somme_premiers` qui, recevant un entier naturel n , renvoie la somme des nombres premiers inférieurs ou égaux à n . Par exemple, `somme_premiers(10)` doit renvoyer 17.

5) Écrire une fonction `premier_suivant` qui, recevant un entier naturel n , renvoie le plus petit nombre premier supérieur ou égal à n . Par exemple, `premier_suivant(8)` doit renvoyer 11.

6) Le premier nombre premier est 2, le deuxième nombre premier est 3, etc. Quel est le millièmème nombre premier ?

7) Tout entier naturel non nul se décompose de manière unique (à l'ordre des facteurs près) en produit de nombres premiers. Par exemple, $84 = 2 \times 2 \times 3 \times 7$. Écrire une fonction `decomposition` qui, recevant un entier naturel n non nul, affiche les nombres premiers qui apparaissent dans la décomposition de n en produit de facteurs premiers.

```
>>> decomposition(84)
2
2
3
7
```