ITC MPSI - 2022-2023 - DS N°1

Durée : 2h

L'utilisation de toute calculatrice, de tout matériel électronique et de tout document est interdite.

On considérera comme acquise toute fonction demandée à une question précédente, même si celle-ci n'a pas été traitée. On indique donc qu'il est toujours possible de faire les dernières questions même si les questions précédentes n'ont pas été traitées. De manière générale, on recommande de faire appel aux fonctions codées au fil du problème.

Problème : Deux conjectures sur les nombres premiers

Introduction

Nous nous intéressons dans ce problème à deux conjectures sur les nombres premiers : la *conjecture des nombres premiers jumeaux* et la *conjecture de Goldbach*. Datant de plusieurs siècles, ces célèbres conjectures faisaient notamment partie de la liste des vingt-trois problèmes de Hilbert, énoncés par ce dernier comme les grands problèmes du XXe siècle lors de la conférence internationale des mathématiciens de 1900. Ces conjectures sont encore non démontrées à ce jour.

Le problème se compose de trois parties :

— La partie I met en place des outils utiles pour la suite. En particulier, il faudra considérer comme acquises les fonctions `is_prime` et `premiers` pour les parties suivantes, quand bien même les questions correspondantes n'ont pas été traitées.

— La partie II porte sur la conjecture des nombres premiers jumeaux.

— La partie III porte sur la conjecture de Goldbach. Elle est indépendante de la partie II.

Partie I : Préliminaires sur les nombres premiers

On rappelle qu'un nombre premier est un entier naturel p qui admet exactement deux diviseurs dans $\mathbb{N}$: 1 et lui-même. On note $\mathcal{P}$ l'ensemble des nombres premiers. La liste des premiers nombres premiers est ainsi : 2, 3, 5, 7, 11, 13, 17, 19, 23 etc.

1. Par définition, un entier n supérieur ou égal à 2 est premier si, et seulement s'il n'admet aucun diviseur d tel que $2 \leq d \leq n-1$. On appelle *test naïf de primalité* de n une procédure consistant à tester sa divisibilité par tous les entiers d compris entre 2 et $n-1$.

Écrire une fonction `is_prime` qui implémente un test naïf de primalité. Elle prendra en argument un entier naturel `n` et renverra le booléen `True` si `n` est premier, et `False` sinon,

2. Le *théorème de Wilson* affirme que n est premier si, et seulement si, n divise $(n-1)!+1$.

a. Écrire une fonction `factorielle` qui prend en argument un entier naturel n et renvoie la valeur de $n!$.

b. Écrire une fonction `Wilson`, utilisant le théorème de Wilson, qui prend en argument un entier naturel `n` et renvoie le booléen `True` si `n` est premier, et `False` sinon.

c. Quelle liste renvoie la commande suivante ?

```
>>> [k for k in range(19) if Wilson(k)]
```

3. Écrire une fonction `premiers` qui prend en argument un entier naturel `N` non nul et renvoie la liste des nombres premiers inférieurs ou égaux à `N`.

Par exemple, on doit obtenir dans la console :

```
>>> premiers(20)
 [2, 3, 5, 7, 11, 13, 17, 19]
```

4. Complétez sur votre copie la fonction `fichier_premiers` ci-dessous, de manière à ce qu'elle prenne en argument un entier naturel `N` non nul et crée un fichier `premiers_N.txt` comportant les `N` premiers nombres premiers, un sur chaque ligne.

```
def fichier_premiers(N):
    f = open(--------,"w")
    f.write(str(2)+"\n") # On écrit 2 sur la première ligne
    c = 1 # compteur des nombres premiers déjà trouvés
    n = 3 # entier actuellement testé
    while c<N:
        if is_prime(n):
            f.write(--------)
            c = --------
        n = --------
    f.close()
    return None
```

Par exemple, l'exécution de `fichier_premiers(50)` crée un fichier `premiers_50.txt` de 50 lignes dont les cinq premières lignes sont :

```
2
3
5
7
11
```

Partie II : Nombres premiers jumeaux

On appelle *couple de nombres premiers jumeaux* un couple $(p, p+2)$ où p et $p+2$ sont des nombres premiers. Par exemple, $(3, 5)$, $(5, 7)$, $(11, 13)$, $(71, 73)$ ou $(881, 883)$ sont des couples de nombres premiers jumeaux. La *conjecture des nombres premiers jumeaux* affirme qu'il existe une infinité de tels couples.

5. Écrire une fonction `jumeaux` qui prend en argument un entier naturel N et renvoie la liste de tous les couples de nombres premiers jumeaux $(p, p+2)$, avec p et $p+2$ inférieurs ou égaux à N.

Par exemple, on doit obtenir dans la console :

```
>>> jumeaux(20)
 [(3, 5), (5, 7), (11, 13), (17, 19)]
```

6. On fait dans cette question l'hypothèse que la conjecture des nombres premiers jumeaux est vraie. Écrire une fonction `cherche_jumeaux` qui prend en argument un entier naturel n et renvoie le couple de nombres premiers jumeaux $(p, p+2)$ avec p minimal tel que $p \geq n$.

Par exemple, on doit obtenir dans la console :

```
>>> cherche_jumeaux(10)
 (11, 13)

>>> cherche_jumeaux(1000)
 (1019, 1021)
```

Partie III : La conjecture de Goldbach

Formulée en 1742 par Christian Goldbach, *la conjecture de Goldbach* affirme que tout nombre entier pair supérieur ou égal à 4 peut s'écrire comme la somme de deux nombres premiers. Par exemple, on a :

$$12 = 5 + 7, \quad 50 = 3 + 47, \quad 128 = 19 + \underbrace{109}_{\in \mathcal{P}}, \quad 1690 = 23 + \underbrace{1667}_{\in \mathcal{P}}.$$

7. Écrire une fonction `Goldbach` qui prend en argument un entier n, pair et supérieur ou égal à 4, et qui renvoie le couple (p, q) de nombres premiers tel que $p + q = n$, avec p minimal si un tel couple existe, et le booléen `False` sinon. Par exemple, on doit obtenir dans la console :

```
>>> Goldbach(1690)
 (23, 1667)
```

8. Écrire une fonction `Gold_test` qui prend en argument un entier naturel N et renvoie le booléen `True` si tous les nombres pairs compris entre 4 et N s'écrivent comme somme de deux nombres premiers, et `False` sinon. Par exemple, on a obtenu dans la console :

```
>>> Gold_test(100000)
 True
```

Ceci signifie que la conjecture de Goldbach est vérifiée jusqu'à 100 000.

9. On suppose dorénavant que la conjecture de Goldbach est vérifiée. Complétez sur votre copie la fonction `Gold_max` ci-dessous, qui prend en argument un entier $N \geq 4$ et renvoie un dictionnaire dont les clés sont les entiers naturels pairs compris entre 4 et N, et la valeur correspondant à la clé n est le couple (p, q) de nombres premiers tel que $p + q = n$, avec p minimal.

```
def Gold_max(N):
    L = [--------] # liste des nombres pairs entre 4 à N
    D = {} # Initialisation du dictionnaire
    for k in L:
        --------
    return --------
```

Par exemple, on doit obtenir dans la console :

```
>>> Gold_max(20)
{4: (2, 2),
 6: (3, 3),
 8: (3, 5),
 10: (3, 7),
 12: (5, 7),
 14: (3, 11),
 16: (3, 13),
 18: (5, 13),
 20: (3, 17)}
```

10. On appelle alors *hauteur* d'un entier naturel n pair supérieur ou égal à 4, le plus petit nombre premier p tel qu'il existe q premier tel que $p + q = n$, c'est-à-dire :

$$H(n) = \min \{p \in \mathcal{P} \mid \exists q \in \mathcal{P},\ p + q = n\}.$$

Par exemple, on a :

$$H(4) = 2, \quad H(6) = H(8) = H(10) = 3, \quad H(12) = 5, \quad H(14) = 3 \ldots$$

Écrire une fonction `H` qui prend en argument un entier n pair supérieur ou égal à 4 et renvoie la valeur de $H(n)$.

On a obtenu à l'aide de cette fonction la représentation graphique suivante, qui montre le comportement «chaotique» de la fonction H et donne ainsi une idée de la complexité de la conjecture de Goldbach.

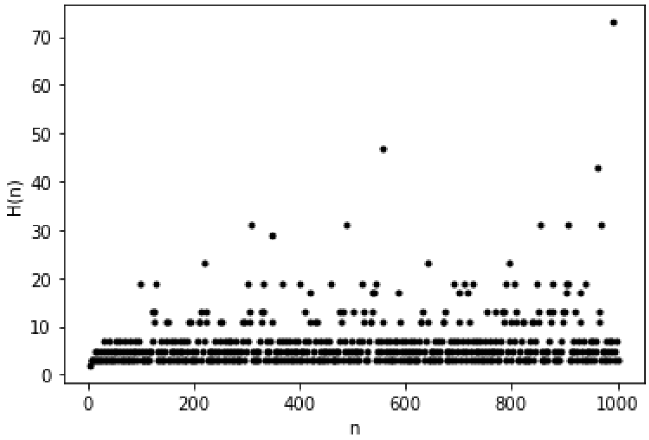

FIGURE 1 – Hauteurs des entiers pairs compris entre 4 et 500.

*** Fin du sujet ***

Remarque : Pour de plus amples informations sur l'état des recherches sur ces conjectures, on pourra consulter :

— https://fr.wikipedia.org/wiki/Nombres_premiers_jumeaux : On y apprend par exemple que les plus grands nombres premiers jumeaux connus, découverts en 2016, possèdent 388 342 chiffres.

— https://fr.wikipedia.org/wiki/Conjecture_de_Goldbach : On y trouve en particulier un tableau qui résume les progrès effectués en direction de cette conjecture, notamment par Terence Tao, arithméticien australien lauréat de la médaille Fields en 2006.