

Extrait du programme de la classe de 2nde (2019)

Utiliser les notions de multiple, diviseur et de nombre premier

Contenus :

- Notations N et Z .
- Définition des notions de multiple, de diviseur, de nombre pair, de nombre impair.

Capacités attendues

- Modéliser et résoudre des problèmes mobilisant les notions de multiple, de diviseur, de nombre pair, de nombre impair, de nombre premier.
- Présenter les résultats fractionnaires sous forme irréductible.

Exemples d'algorithmes

- Déterminer si un entier naturel a est multiple d'un entier naturel b .
 - Pour des entiers a et b donnés, déterminer le plus grand multiple de a inférieur ou égal à b .
 - Déterminer si un entier naturel est premier.
-

Quelques programmes autour des nombres premiers

1. Proposer un programme Python qui permet de déterminer si un entier naturel est premier.
2. Voici 4 programmes qui répondent à la consigne précédente.
Proposer des critères pour comparer ces programmes. Lesquels proposeriez-vous à des élèves de seconde ? Avec quelles éventuelles modifications ? Dans quelles situations ?

Programme 1¹

```
1 def est_premier1(nombre):
2     ntest = 2
3     reponse = True
4     while ntest < nombre and reponse :
5         if nombre % ntest == 0 :
6             reponse = False
7         ntest = ntest + 1
8     return reponse
```

Programme 2²

```
1 def est_premier2(n):
2     diviseurs=[]
3     for i in range(1, n+1):
4         if n % i == 0:
5             diviseurs.append(i)
6     if len(diviseurs) == 2 :
7         return True
8     else:
9         return False
```

Programme 3³

```
1 def est_multiple(a,b):
2     return a % b == 0
3
4 def est_premier3(n):
5     for d in range(2, n):
6         if est_multiple(n, d):
7             return False
8     return n>1
```

Programme 4

```
1 from math import sqrt
2 def est_premier4(n):
3     if n % 2 == 0:
4         return False
5     maxi = int(sqrt(n))
6     for d in range(3, maxi + 1, 2):
7         if n % d == 0:
8             return False
9     return True
```

3. Le module *time* contient la fonction `time()` qui renvoie le temps écoulé en secondes depuis le 01/01/1970.
Par différence entre deux appels, on peut déterminer approximativement le temps nécessaire pour réaliser une ou plusieurs instructions.

```
from time import time
def temps_ecoule(n):
    debut = time()
    resultat = est_premier...(n)
    fin = time()
    return fin-debut
```

Utiliser la fonction `temps_ecoule(n)` pour comparer les temps d'exécution des quatre fonctions pour les valeurs de n : 15 485 863 et 179 424 673.

1. d'après Le livre scolaire

2. d'après Benjamin Clerc sur Mathématique

3. d'après Alain Busser sur Mathématique

4. Voici 4 questions qui utilisent la notion de nombre premier. Pour répondre à chacune de ces questions, quel(s) programme(s) Python est-il pertinent d'utiliser / d'adapter ?
- (a) 2021 est-il premier ?
 - (b) 15 485 863 est-il premier ?
 - (c) Donner tous les nombres premiers inférieurs à 1000.
 - (d) Deux nombres premiers jumeaux sont deux nombres premiers qui ne diffèrent que de 2. Quels sont les plus grands nombres premiers jumeaux inférieurs à 1000 ? à 1 000 000 000 ?