MR JÉAN-PAUL CARDINAL

COMPTÉ RÉNDU
TRAVAUX DIRIGÉS 4

ÉTUDIANT : MOHAMMÉD LAHJAJI

Introduction:

L'étude de l'arithmétique remonte à Euclide. On y rencontre les noms de pierre de Fermat, Leonhardt Euler , Karl Friedrich Gauss , Étienne Bézout , Jacques Hadamard, et c'est encore actuellement un domaine de recherche très actif.

Exercice 1: Un nombre est-il premier ?

a) Soient `a` et `b` deux entiers, le quotient de la division euclidienne de `a` par `b` est : `(a/b)` et

le reste R de cette division est : `R = a%b`.

b) Un entier p strictement supérieur à 1 est dit premier , si ses seuls diviseurs positifs sont 1 et p.

c) les entiers 2017, 3001, 49999 sont premiers mais les entiers 1001 et 89999 ne sont pas premier car 1001 = 7*143 et 89999 = 7*12857

d) La fonction `is_prime` est :

```
import random
import numpy as np
import math
```

La description de `is_prime` est la suivante :

```
def is_prime(n):
    if (n < 2):
        return False

    for p in range(2, (int(np.sqrt(n))+1)):
        if n%p == 0:
            return False

    return True
```

e) soit la fonction Fn suivante :

```
def Fn(n):
    return 2**(2**n) + 1
```

D'où pour `F0,F1,F2,F3,F4,F5 :`

le nombre de Fermat 1 est premier .
le nombre de Fermat 5 est premier
le nombre de Fermat 17 est premier
le nombre de Fermat 257 est premier
le nombre de Fermat 65537 est premier
le nombre de Fermat 4294967297 n est pas premier

Exercice 2: crible d'erasthotène , distribution des nombres premiers

a) D'après le crible d'Ératosthène :

Les nombres premiers inférieurs à 200 sont :
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199.

b) La fonction primes est :

```
def primes(n):
    P = [k for k in range(2, n) if is_prime(k)]
    return P
```

c) la listes des nombres premiers inférieurs à 1000 :

```
P = primes(1000)
 with open('primes','w') as f:
    f.write('Tableau contenant tous les entiers de 1 a 1000\n')
    cpt = 0
    for p in P:
          cpt += 1
          f.write('{0}  '.format(p))
          if cpt%10 == 0:
                f.write('\n')
```

Exercice 3: Factorisation d'un entier en premiers.

a) Tout entier n supérieur strictement à 1 est un produit de facteurs premiers.
La décomposition est unique à l'ordre prés.

b) $924 = 2^2*3*7*11$

c) Voici la fonction Factors :

```
def factors(n):
    P = primes(n)
    FF = []
    for p in P:
          while n%p == 0:
                n = n/p
                FF.append(p)
    return FF
```

Exercice 4: PGCD de deux entiers , identité de Bézout , algorithme d'Euclide

a) soient a et b deux entiers l'ensembles des diviseurs communs de a et b est l'ensemble des diviseurs d'un unique entier positif , appelé P.G.C.D

b) on utulisant la fonction pgcd on peut calculer led PGCD de 4864 et 3458 :

```
def pgcd(a, b):
    r = a%b
    while r != 0:
        a = b
        b = r
        r = a%b
    return b
```

On trouve que PGCD est 38

b) Deux entiers sont premiers entre eux si leurs PGCD est 1. Le théorème de Bézout est : deux entiers a et b sont premiers entre eux si et selement s'il existent deux entiers u et v rels que au+bv=1

c) l'algorithme d'Euclide est un algorithme qui permet la détermination du PGCD de deux entiers , Dans l'algorithme d'Euclide le PGCD de deux entiers correspond au dernier reste non nul .

d) A partir de l'algorithme d'Euclide , Diophante un mathématicien grec de l'école d'Alexandrie a inventé les équations diophantiennes qui sont sous la forme xa+yb=d avec x,y,d appartient à Z

On a : 4864 = 3458+ 1406
3458 = 1406*2 + 646
1406 = 646*2 + 114
646 = 114 * 5 + 76
114 = 76 + 38
76 = 38*1 + 0
D'où le PGCD est 38

A partir de l'équation diophantienne on déduit le couple (x,y)

Or 32*4862 – 45*3458 = 38
Et x*4864 + y* 3458 = 38
Donc :
4864*(x+32)=(y+45)*3458
On simplifie par 38 on trouve : 128*(x+32) = (y+45)*91
Or 128 divise (y+45)*91 et premier avec 91 donc il existe k appartient à z tels que :
y+45 = 128*k . de même pour 91 on trouve qu'il existe un k' appartient à Z tels que :
x+32=91*k'
d'où : (x,y) = (91k'-32 , 128*k-45)

e) La fonction Euclide :

```
def euclide(a,b):
     d = a
     dp = b
     x = 1
     y = 0
     xp = 0
     yp = 1
     while (dp != 0):
          q = d/dp
          ds = d
          xs = x
          ys = y
          d = dp
          x = xp
          y = yp
          dp = ds - q*dp
          xp = xs - q*xp
          yp = ys - q*yp
     return  d,x,y
```

Conclusion:

Ce TP m’a permis de faire des recherches sur beaucoup de fonctions qui permettent la détermination du PGCD et les coefficient de l’équation diophantienne ainsi de connaitre l’utilité de ces fonctions dans le domaine de l’arithmétique .