Sommes par lignes d'une matrice et nombres premiers (Denise Vella-Chemla, 23.9.2018)

Soit la matrice M_n suivante, qu'on imagine de taille infinie mais qui sera de taille finie pour les exemples founis ici, et qui contient des 1 en première colonne et dans les autres colonnes, en position $M[i,j]$ la plus grande puissance de j qui divise i.

Fournissons l'exemple d'une telle matrice pour $n = 10$.

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 3 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Effectuer les sommes par lignes de M_n fournit une matrice d'une seule colonne dont tous les éléments égaux à 2 sont d'indice un nombre premier. Le programme python qui code cette idée est très court et il semble efficace.

```python
import numpy as np

def pgp(x, y):
  e = 0
  while (x/y > 0):
    x = x / y
    e = e + 1
  return e

def elt(x, y):
  return 1 if y == 1 else (pgp(x, y) if x % y == 0 else 0)

import time
tmps1=time.time()
n = input("Entrer n : ")
m = np.array([[elt(x, y) for y in range(1, n+1)] for x in range(1, n+1)])
c1 = np.sum(m, axis=1)
p = [i+1 for i in range(c1.size) if c1[i] == 2]
print(p)
print(len(p))
print(c1.size)
tmps2=time.time()-tmps1
print "Temps d'execution avec matrices = %f" %tmps2
```

Voici la table des sommes par lignes pour les entiers de 1 à 100.

lod(1) = 1	lod(21) = 4	lod(41) = 2	lod(61) = 2	lod(81) = 9
lod(2) = 2	lod(22) = 4	lod(42) = 8	lod(62) = 4	lod(82) = 4
lod(3) = 2	lod(23) = 2	lod(43) = 2	lod(63) = 7	lod(83) = 2
lod(4) = 4	lod(24) = 10	lod(44) = 7	lod(64) = 15	lod(84) = 13
lod(5) = 2	lod(25) = 4	lod(45) = 7	lod(65) = 4	lod(85) = 4
lod(6) = 4	lod(26) = 4	lod(46) = 4	lod(66) = 8	lod(86) = 4
lod(7) = 2	lod(27) = 6	lod(47) = 2	lod(67) = 2	lod(87) = 4
lod(8) = 6	lod(28) = 7	lod(48) = 4	lod(68) = 7	lod(88) = 10
lod(9) = 4	lod(29) = 2	lod(49) = 4	lod(69) = 4	lod(89) = 2
lod(10) = 4	lod(30) = 8	lod(50) = 7	lod(70) = 8	lod(90) = 13
lod(11) = 2	lod(31) = 2	lod(51) = 4	lod(71) = 2	lod(91) = 4
lod(12) = 7	lod(32) = 11	lod(52) = 7	lod(72) = 16	lod(92) = 7
lod(13) = 2	lod(33) = 4	lod(53) = 2	lod(73) = 2	lod(93) = 4
lod(14) = 4	lod(34) = 4	lod(54) = 10	lod(74) = 4	lod(94) = 4
lod(15) = 4	lod(35) = 4	lod(55) = 4	lod(75) = 7	lod(95) = 4
lod(16) = 9	lod(36) = 12	lod(56) = 10	lod(76) = 7	lod(96) = 17
lod(17) = 2	lod(37) = 2	lod(57) = 4	lod(77) = 4	lod(97) = 2
lod(18) = 7	lod(38) = 4	lod(58) = 4	lod(78) = 8	lod(98) = 7
lod(19) = 2	lod(39) = 4	lod(59) = 2	lod(79) = 2	lod(99) = 7
lod(20) = 7	lod(40) = 10	lod(60) = 13	lod(80) = 14	lod(100) = 12

La définition des éléments de la matrice M qu'on a utilisée est :

$$\begin{array}{lll} \forall y \in \mathbb{N}, & M[1,y] & = 1 & \\ \forall x > 1, y \in \mathbb{N}, & M[yx,y] & = 0 & \textit{si y ne divise pas x} \ \ (\text{y} \not| \, x) \\ & & = M[x,y]+1 & \textit{sinon} \end{array}$$

Si on remplace dans le programme l'appel à la fonction $pgp(x,y)$ (qui fournit la plus grande puissance de y divisant x) par 1, on peut définir ainsi la valeur des éléments de la matrice P :

$$\begin{array}{llll} \forall x \in \mathbb{N}, & P[x,x] & = 1 & \\ \forall x, y \in \mathbb{N} \times \mathbb{N}, & P[x,y] & = 0 & \textit{si } y > x \\ & & = P[x-y,y] & \textit{sinon.} \end{array}$$

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

En calculant les sommes par lignes de cette matrice, on obtient comme attendu ligne par ligne le nombre de diviseurs de chaque entier et à nouveau, les éléments d'image 2 sont les nombres premiers :

nbdiv(1) = 1	nbdiv(21) = 4	nbdiv(41) = 2	nbdiv(61) = 2	nbdiv(81) = 5
nbdiv(2) = 2	nbdiv(22) = 4	nbdiv(42) = 8	nbdiv(62) = 4	nbdiv(82) = 4
nbdiv(3) = 2	nbdiv(23) = 2	nbdiv(43) = 2	nbdiv(63) = 6	nbdiv(83) = 2
nbdiv(4) = 3	nbdiv(24) = 8	nbdiv(44) = 6	nbdiv(64) = 7	nbdiv(84) = 12
nbdiv(5) = 2	nbdiv(25) = 3	nbdiv(45) = 6	nbdiv(65) = 4	nbdiv(85) = 4
nbdiv(6) = 4	nbdiv(26) = 4	nbdiv(46) = 4	nbdiv(66) = 8	nbdiv(86) = 4
nbdiv(7) = 2	nbdiv(27) = 4	nbdiv(47) = 2	nbdiv(67) = 2	nbdiv(87) = 4
nbdiv(8) = 4	nbdiv(28) = 6	nbdiv(48) = 10	nbdiv(68) = 6	nbdiv(88) = 8
nbdiv(9) = 3	nbdiv(29) = 2	nbdiv(49) = 3	nbdiv(69) = 4	nbdiv(89) = 2
nbdiv(10) = 4	nbdiv(30) = 8	nbdiv(50) = 6	nbdiv(70) = 8	nbdiv(90) = 12
nbdiv(11) = 2	nbdiv(31) = 2	nbdiv(51) = 4	nbdiv(71) = 2	nbdiv(91) = 4
nbdiv(12) = 6	nbdiv(32) = 6	nbdiv(52) = 6	nbdiv(72) = 12	nbdiv(92) = 6
nbdiv(13) = 2	nbdiv(33) = 4	nbdiv(53) = 2	nbdiv(73) = 2	nbdiv(93) = 4
nbdiv(14) = 4	nbdiv(34) = 4	nbdiv(54) = 8	nbdiv(74) = 4	nbdiv(94) = 4
nbdiv(15) = 4	nbdiv(35) = 4	nbdiv(55) = 4	nbdiv(75) = 6	nbdiv(95) = 4
nbdiv(16) = 5	nbdiv(36) = 9	nbdiv(56) = 8	nbdiv(76) = 6	nbdiv(96) = 12
nbdiv(17) = 2	nbdiv(37) = 2	nbdiv(57) = 4	nbdiv(77) = 4	nbdiv(97) = 2
nbdiv(18) = 6	nbdiv(38) = 4	nbdiv(58) = 4	nbdiv(78) = 8	nbdiv(98) = 6
nbdiv(19) = 2	nbdiv(39) = 4	nbdiv(59) = 2	nbdiv(79) = 2	nbdiv(99) = 6
nbdiv(20) = 6	nbdiv(40) = 8	nbdiv(60) = 12	nbdiv(80) = 10	nbdiv(100) = 9