

nphbasic_5a.py

```
1  #!/usr/bin/env python3.6
2  # -*- coding: utf-8 -*-
3  # Série nphbasic_1a.Général Python
4  # Version nphb_5a.py
5
6  import time
7  from math import sqrt
8
9
10 def cosmic(un_o):
11     """
12     La multiplication est conditionnelle au typage.
13     """
14     debut4 = time.time()
15     tiwp = [] # Prélèvement des primeures
16     towop = {} # Dictionnaire principal
17     quo = un_o
18     sqrip = int(sqrt(quo)) # Racine première
19     sq3 = int(sqrt(sqrip)) # Racine médiane
20     sq4 = int(sqrt(sq3)) # Racine minimale
21
22     # La principale fonction itérative
23     def gennau():
24         # La technique de recherche primeure
25         dvs = quo
26         ipo = dvs % 6
27         ip = 0
28         for ip in range(2, sq3):
29             if ip % 6 in (1, 5) and not quo % ip:
30                 dvs = int(quo / ip) # dvs = Second niveau
31                 break
32         else: # Recherche plan minimal
33             for ip in range(2, sq4):
34                 ouq = int(quo / ip)
35                 if ouq % 6 in (5, ipo) and not quo % ip:
36                     dvs = ouq
37                     break
38     print(ip, 'dvs', dvs, 'sq3', sq3, 'sq4', sq4)
39
40     # L'initialisation des tables factorielles
41     sqrwp = int(sqrt(dvs)) # Racine secondaire
42     wp = 1
43     while wp < sqrwp + 1:
44         if not dvs % wp: # Nombre multiple
45             wyp = wp, int(dvs / wp)
46             tiwp.append(wyp) # Prélèvement
47             wp += 1
48         else: # Nombre non multiple
49             while wp < sqrwp + 1:
50                 if not dvs % wp:
51                     break
52                 wp += 1
53
54     # La reconstitution initiale
55     for zp in tiwp:
56         zop = zp # Prélèvement
57         multinf = zop[0] # Multiple inférieur
58         multsup = zop[1] # Multiple supérieur
```

```

59         z1_10 = multinf # Primeure absente
60         m1 = int(quo / multinf)
61         if multsup > sqrip: # Test supérieur
62             m2_10 = int(quo / multsup)
63         else: # Test inférieur
64             m2_10 = multsup
65             z2_10 = m2_10
66             m2 = int(quo / z2_10)
67             towop[z2_10] = m2
68             towop[z1_10] = m1
69
70         # La reconstitution terminale
71         for ap, op in sorted(towop.items()): # Avant transmission
72             yield ap, op, ap % 6, op % 6
73         print('eleme', len(towop), 'typ', un_o % 6, 'sqrip', sqrip)
74
75         # Seuil de la transmission
76         nau = gennau()
77         gers = ''
78         for g in nau:
79             gers += ': (n) {} * {} typ {}*{}\n'.format(g[0], g[1], g[2],
g[3])
80         print('{}nphb_5a.py En : {}'.format(gers, time.time() - debut4))
81
82
83     # Début
84     unO = 987654321234
85     print('Cosmic')
86     cosmic(unO)
87     """
88     Mémo 12345678987654321
89     """

```