

<p style="text-align: center;">Informatique en CPGE (2017-2018) Exercices : nombres</p>

1.

```
def divise(p, q) :  
    return q%p==0
```
2.

```
def estpremier(p) :  
    q=2  
    while q<p:  
        if divise(q,p) : return 0  
        q+=1  
    if p>1: return 1  
    else: return 0
```

3. Un code (loin d'être efficace) possible est :

```
def phi(p) :  
    n, q=0, 2  
    while q<=p:  
        n+=estpremier(q)  
        q+=1  
    return n
```

4. (a) Deux suites (a_n) et (b_n) à termes non nuls sont dite équivalentes si par définition

$$\lim_{n \rightarrow +\infty} \frac{a_n}{b_n} = 1$$

(b) Le théorème des nombres premiers implique que $\lim_{n \rightarrow +\infty} \varphi(n) = \lim_{n \rightarrow +\infty} \frac{n}{\ln(n)} = +\infty$.

D'où l'existence d'une infinité de nombres premiers.

(c) Par exemple :

```
def theta(N) :  
    from math import log  
    return abs((phi(N)*log(N))/N - 1)
```

```
def test(epsilon) :  
    N=50  
    while True:  
        if theta(N)<=epsilon: return N  
        N+=1
```

(d) On peut simplement écrire :

```
import matplotlib.pyplot as plt  
X=[i for i in range(50, 5001)]  
Y=[theta(i) for i in X]  
plt.plot(X, Y)  
plt.show()
```

Ce qui donne :

