Exercice : Algorithme de Karatsuba pour multiplier deux polynômes.

$$P = \sum_{k=0}^{n} a_k . X^k$$

P sera représenté informatiquement par $[a_0, a_1, \dots, a_n]$.
Par exemple $P = X$ peut être représenté par $[0, 1]$ ou $[0, 1, 0, 0]$.

1)

```
def add(p, q):
    dp = len(p)-1
    dq = len(q)-1
    n  = max(dp, dq)+1
    res = [0 for k in range(n)]
    for k in range(len(p)):
        res[k] = res[k] + p[k]
    for k in range(len(q)):
        res[k] = res[k] + q[k]
    return res
```

Il y a donc $len(p) + len(q)$ additions. Donc la complexité est en

$$\underset{len(p), len(q) \infty}{O}(len(p) + len(q))$$

2)

```
def diff(p, q):
    moins_q = [-a for a in q]
    return add(p, moins_q)
```

Il y a : $len\, q + len(p) + len(q)$ opérations.
$= len(p) + 2\, len(q)$

On a une complexité en $O(len(p) + len(q))$

3)

```
def multiplier(p, q):
    dp = len(p)-1
    dq = len(q)-1
    dr = dp + dq
    res = [0 for k in range(dr+1)]
    for k in range(dp+1):
        for i in range(dq+1):
            res[i+k] = res[i+k] + p[k].q[i]
    return res
```

La complexité est en : $2\, len(p).len(q)$. La complexité est donc en.

$$O(len(p)\, len(q))$$

$$P = \sum_{k=0}^{n} a_k . X^k \qquad Q = \sum_{i=0}^{m} b_i . X^i .$$

$$P.Q = \sum_{k=0}^{n} \sum_{i=0}^{m} a_k . b_i \, X^{i+k} .$$

4a) Soit P et $Q \in \mathbb{R}_n[X]$. Soit $k \in [\![0, n]\!]$. On effectue la division euclidienne de P et Q par X^k. Il existe donc $P_1, Q_1 \in \mathbb{R}[X]$ et $P_0, Q_0 \in \mathbb{R}_{k-1}[X]$ tels que.

$$P = X^k P_1 + P_0$$
$$Q = X^k Q_1 + Q_0$$

$$\begin{aligned} P.Q &= (X^k P_1 + P_0)(X^k Q_1 + Q_0) \\ &= X^{2k} . P_1 Q_1 + X^k (P_1 Q_0 + P_0 Q_1) + P_0 Q_0 \\ &= X^{2k} . P_1 Q_1 + X^k \left((P_1 + P_0)(Q_1 + Q_0) - P_1 Q_1 - P_0 Q_0 \right) + P_0 Q_0 \end{aligned}$$

4b) On suppose que $n \in \mathbb{N}^*$ et que $k = \left\lfloor \frac{n+1}{2} \right\rfloor$.

- $\deg P_0 < k$ et $P = X^k P_1 + P_0$. Donc $\deg (X^k P_1) \leqslant \deg (P)$.
  Donc $\deg (P_1) + k \leqslant \deg P$ donc $\deg P_1 \leqslant \deg P - k$
  $\leqslant n - k$
  $\leqslant n - \left\lfloor \frac{n+1}{2} \right\rfloor$

  De même $\deg Q_1 \leqslant n - \left\lfloor \frac{n+1}{2} \right\rfloor$

  Montrons que $\deg P_1 \leqslant \left\lceil \frac{n-1}{2} \right\rceil$

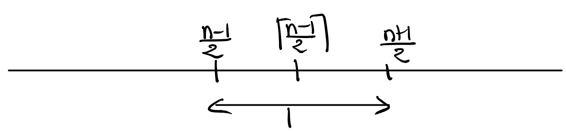
- $$\left\lfloor \frac{n+1}{2} \right\rfloor \leqslant \frac{n+1}{2} < \left\lfloor \frac{n+1}{2} \right\rfloor + 1 \qquad (1)$$

$$\left\lceil \frac{n-1}{2} \right\rceil - 1 < \frac{n-1}{2} \leqslant \left\lceil \frac{n-1}{2} \right\rceil \qquad (2)$$

Donc $\left\lfloor \frac{n+1}{2} \right\rfloor > \frac{n+1}{2} - 1$ d'après (1)

donc $- \left\lfloor \frac{n+1}{2} \right\rfloor < - \frac{n-1}{2}$

donc $n - \left\lfloor \frac{n+1}{2} \right\rfloor < n - \frac{n-1}{2} = \frac{n+1}{2}$.



Donc $\left\lceil \frac{n-1}{2} \right\rceil$ est le plus grand entier strictement inférieur à $(n+1)/2$. Donc, comme $n - \left\lfloor \frac{n+1}{2} \right\rfloor$ est un entier.

$$n - \left\lfloor \frac{n+1}{2} \right\rfloor \leqslant \left\lceil \frac{n-1}{2} \right\rceil$$

On a en effet $\underbrace{n - \left\lfloor \frac{n+1}{2} \right\rfloor}_{\in \mathbb{Z}} < \frac{n+1}{2} = \frac{n-1}{2} + 1$

$< \underbrace{\left\lceil \frac{n-1}{2} \right\rceil + 1}_{\in \mathbb{Z}}$

Donc $$n - \left\lfloor \frac{n+1}{2} \right\rfloor \leqslant \left\lceil \frac{n-1}{2} \right\rceil$$

<u>Corrigé</u> : ~~$n - \left\lfloor \frac{n+1}{2} \right\rfloor = \frac{n-1}{2}$ $\quad n=4 \quad 4 - \left\lfloor \frac{5}{2} \right\rfloor = \frac{3}{2}$, $4 - 2 = \frac{3}{2}$, $2 = \frac{3}{2}$~~

Donc $\deg P_1 \leqslant \left\lceil \frac{n-1}{2} \right\rceil$ et $\deg Q_1 \leqslant \left\lceil \frac{n-1}{2} \right\rceil$

- $\deg P_0 \leqslant k-1$ et $\deg Q_0 \leqslant k-1$. Montrons que $k-1 \leqslant \left\lceil \frac{n-1}{2} \right\rceil$ càd que :

$$\left\lfloor \frac{n+1}{2} \right\rfloor - 1 \leqslant \left\lceil \frac{n-1}{2} \right\rceil$$

En effet

$$\left\lfloor \frac{n+1}{2} \right\rfloor \leqslant \frac{n+1}{2} \quad \text{donc} \quad \left\lfloor \frac{n+1}{2} \right\rfloor - 1 \leqslant \frac{n-1}{2} \leqslant \left\lceil \frac{n-1}{2} \right\rceil$$

- $\deg(P_0 + P_1) \leqslant \max(\deg(P_0), \deg(P_1)) \leqslant \left\lceil \frac{n-1}{2} \right\rceil$.

De même $\deg(Q_0 + Q_1) \leqslant \left\lceil \frac{n-1}{2} \right\rceil$

Donc

$$PQ = \underset{\text{Trivial}}{X^{2k}} \cdot P_1Q_1 + \underset{\text{Trivial}}{X^{k}} \left((P_1 + P_0)(Q_1 + Q_0) - P_1Q_1 - P_0Q_0 \right) + P_0Q_0$$

4.c)
```
def decoupage(P, k):
    p0 = [p[i] for i in range(k)]
    p1 = [p[i] for i in range(k, len(p))]
    return p0, p1
```

4.d)
```
def Karatsuba(p, q)
    if len(p) <= 1 and len(q) <= 1 :
        return [ p[0]*q[0]]
    else
        dp = len(p)-1
        dq = len(q)-1
        n = max(dp, dq)
        k = (n+1)//2
        p0, p1 = decoupage(p, k)
        q0, q1 = decoupage(q, k)
        p1q1 = Karatsuba(p1, q1)
        p0q0 = Karatsuba(p0, q0)
        u = Karatsuba( add(p1, p0), add(q1, q0))
        v = add(p1q1, p0q0)
        u = diff(u, v)
        np1q1 = [0 for i in range(2k)] + p1q1
        nu = [0 for i in range(k)] + u
        res = add(np1q1, nu)
        res = add(res, p0q0)
        return res.
```

4.e)

Mines - Ponts 2019

Q1) Réponse d'ingénieur :
```
import math as ma.
ma.log(0.5).
```

Réponse attendue au concours :
```
from math import *
log(0.5)
```

Q2)
```
def sont_proches(x, y)
    atol = 1.0e-5
    rtol = 1.0e-8
    return abs(x-y) <= atol + rtol*abs(y)
```

Q3)
$$\begin{aligned} \text{mystere}(1001, 10) &= 1 + \text{mystere}(100.1, 10) \\ &= 1 + 1 + \text{mystere}(10.01, 10) \\ &= 1 + 1 + 1 + \underbrace{\text{mystere}(1.001, 10)}_{=0} \\ &= 3. \end{aligned}$$

Q4) Si $x \geqslant 1$ et $b \geqslant 2$ $\quad \dfrac{x}{b^n} < b \qquad \dfrac{x}{b^{n-1}} \geqslant b$

donc $\quad 1 \leqslant \frac{x}{b^n} < b$

donc $\quad b^n \leqslant x < b^{n+1}$

donc $\quad n \ln b \leqslant \ln x \leqslant (n+1) \ln b$

donc $\quad n \leqslant \frac{\ln x}{\ln b} < n+1$

Donc $\quad n = \left\lfloor \frac{\ln(x)}{\ln(b)} \right\rfloor = \lfloor \log_b(x) \rfloor$

Donc mystere(x, b) renvoie $\lfloor \log_b(x) \rfloor$

Q5) Le calcul de x2 accumule les erreurs d'arrondi de 10^5 additions. Il est donc moins précis que le calcul de x_1 qui se fait en une seule opération.

II.

Q6) 4 Go $\simeq 4 \cdot 10^9$ octets.

1 ko $\simeq 10^3$ octets.
1 Mo $\simeq 10^3$ ko $\simeq 10^6$ octets
1 Go $\simeq 10^3$ Mo $\simeq 10^9$ octets.

Or 32 bits $= 4 \times 8$ bits.
$= 4$ octets.

Donc $\frac{4 \text{ Go}}{4\text{o}} \simeq 10^9$

Donc on peut stocker une liste de booléen de longueur 10^9.

Q7) Un booléen peut prendre 2 valeurs. On peut donc le coder sur 1 bit. Donc on peut gagner un facteur 32.

Q8)