# TP 9 : Comparaison d'algorithmes

L'objectif de ce TP est d'utiliser les techniques de calculs de complexité étudiées dans le cours sur un exemple : l'évaluation polynomiale.

Soit $n \in \mathbb{N}$. Soit $P \in \mathbb{R}_n[X]$. Alors il existe $(a_0, \dots, a_n) \in \mathbb{R}^{n+1}$ tel que :

$$P(X) \;=\; \sum_{k=0}^{n} a_k \, X^k$$

On stocke les coefficients du polynôme $P$ dans une liste `a` :

$$\texttt{a} \;=\; [a_0, \dots, a_n]$$

## I. Algorithme naïf

On propose une première fonction pour évaluer le polynôme $P$ en un réel $x$.

```
def polynome(x, a) :
    n = len(a) - 1
    S = 0
    for k in range(n+1) :
        puissance = 1
        for i in range(k) :
            puissance = puissance * x
        S = S + a[k] * puissance
    return S
```

- Donner la spécification de cet algorithme.
- Démontrer la terminaison de cet algorithme.
- Complexité.
  - Déterminer le nombre total de multiplications et d'additions effectués en fonction de $n$.
  - En déduire la complexité de cet algorithme.

## II. Algorithme de Horner

Le second algorithme, appelé algorithme de Horner, consiste à effectuer le calcul de $P(x)$ en exploitant l'égalité suivante :

$$P(x) \;=\; \Big(\big((a_n \, x + a_{n-1}) \, x + a_{n-2}\big) \, x + \cdots + a_1\Big) \, x + a_0$$

- Écrire, en donnant sa spécification, une fonction `Horner` qui prend en paramètre la liste `a` des coefficients d'un polynôme $P$ et un réel `x`, et qui renvoie l'évaluation de $P$ en `x` à l'aide de l'algorithme de Horner.
- Démontrer la terminaison de cet algorithme.
- Complexité.
  - Déterminer le nombre total de multiplications et d'additions effectuées en fonction de $n$.
  - En déduire la complexité de ce nouvel algorithme.