TP n° 3 – Interpolation polynomiale et base de Newton

Introduction

Le but de ce TP est d'explorer variante de la méthode directe afin d'interpoler une fonction par un polynôme vue lors du TP n° 1.

Soit :

- $a = x_0 < x_1 < \ldots < x_n = b$ une subdivision d'un intervalle $[a\,;\,b]$;
- f une fonction dont on connaît les valeurs $y_i = f(x_i)$, pour i allant de 0 à n.

Objectif

On cherche un polynôme P de degré au maximum égal à n et tel que :

$$y_i = P(x_i), \text{ pour } 0 \text{ allant de } 0 \text{ à } n$$

On travaille dans la base de Newton $(N_0, N_1, N_2, \ldots, N_n)$ avec $N_0 = 1$ et :

$$\forall k \in [\![1\,;\,n]\!], \quad N_k = (x - x_0)(x - x - 1)\ldots(x - x_{k-1})$$

On cherche P sous la forme :

$$P = a_0 + a_1.N_1 + a_2.N_2 + \ldots + a_n.N_n$$

Questions

1. Après avoir traduit $y_i = P(x_i)$, pour i allant de 0 à n, écrire ces égalités sous forme d'un système triangulaire.
2. Rappeler la résolution par différences divisées.
3. Tracer sur un même graphique une fonction f $\left(f = sin \text{ puis } f = \dfrac{1}{1+10x^2}\right)$ ainsi que son polynôme d'interpolation associé.

Attendus

Chaque binôme devra rendre :

- Un compte rendu succint (suivre l'énoncé pour le plan) ;
- Un code (Python) qui fonctionne.

Indications et contenu souhaitable :

1. Entrées : a, b, n et f (par exemple $f = sin$ sur $[0\,;\,2\pi]$, $n = 10, 20\ldots$).
2. Ecrire une fonction Base qui prend en paramètres X, k, x et qui renvoie $L_k(x)$.
3. Ecrire une fonction qui évalue le polynôme de Newton en x réel donné.
4. Passer à l'affichage en choisissant le nombre de points (au-dessus de 500) pour tracer f et P.