

## Interpolation de Lagrange :

Le problème est de déterminer l'unique polynôme de degré  $n$ ,  $P_n$  qui satisfait

$$P_n(x_i) = f(x_i), \quad \forall i = 0, \dots, n. \quad (1)$$

Le polynôme qui satisfait cette égalité le polynôme d'interpolation de Lagrange

$$P_n(x) = \sum_{k=0}^n l_k(x) f(x_k), \quad (2)$$

où les  $l_k$  sont des polynômes de degré  $n$  qui forment une base de  $\mathcal{P}_n$  :

$$l_k(x) = \prod_{i=0, i \neq k}^n \frac{x - x_i}{x_k - x_i}. \quad (3)$$

---

Pour mon programme, je choisis de mettre la liste des abscisses (les  $x_i$  de ce qui est au-dessus) dans une liste nommée  $\theta_x$ , les ordonnées (donc les  $f(x_i)$  de ce qui est au-dessus) dans une liste nommée  $\theta_y$  et la variable souhaitée pour la polynôme de Lagrange ( $x$  dans ce qui est au-dessus) sera désignée par  $\theta_v$  ( $v$  comme variable !).

Les lignes même correspondent à du calcul matriciel, car la TI-Voyage 200 est capable d'effectuer plusieurs calcul en même temps :

- La première ligne du FOR sert à calculer d'un seul coup chaque fraction de l'équation 3 (donc pour tous les  $i$  entre 0 et  $n$  tels que  $i \neq k$ ) pour  $\theta_k = 1$ . On obtient donc dans  $\theta_r$  non pas le polynôme  $l_{\theta_k}(\theta_v)$  mais la liste à  $n$  éléments  $\left( \frac{x - x_i}{x_k - x_i} \right)_{i=0, \dots, n | i \neq k}$ .
- La calculatrice ne sachant pas calculer le produit de 0 à  $n$  sans compter l'élément  $k$ , la deuxième ligne sert à définir le  $k$ -ième élément de cette liste (qui dans la calculatrice n'existe pour l'instant pas) comme étant le nombre 1.
- Le « product( $\theta_r$ ) » effectue alors le produit de tous les éléments de la liste  $\theta_r$ , ce sera donc le polynôme  $l_{\theta_k}$  de l'équation 3. Le  $\theta_y[\theta_k]$  représente clairement le  $k$ -ième élément de la liste des ordonnées, c'est-à-dire en théorie  $f(x_k)$ . Le produit des deux donne donc l'élément  $l_k(x) f(x_k)$  de l'équation 2.
- La dernière ligne du FOR est un procédé récursif (enfin c'est comme ça que ça s'appelle). Par exemple, pour calculer la somme de  $f(n)$  pour  $n$  allant de 1 à 10, on crée une variable temporaire  $s$  (pour somme) qu'on initialise à 0 (à 1 si ça avait été une multiplication), et on fait une boucle For agissant sur  $n$  allant de 1 à 10, dans laquelle la seule ligne sera  $s + f(n) \rightarrow s$ . À la sortie de cette boucle, la variable  $s$  contiendra donc la valeur voulue. C'est le même procédé employé ici pour calculer la somme de l'équation 2. Le résultat dans  $\theta_p$  à la sortie du For est donc le polynôme recherché !!!
- Enfin, une fonction dans la TI-Voyage 200 renvoie automatiquement la dernière valeur calculée, qui est justement ici  $\theta_p$ , donc inutile de le stipuler.

---

En ce qui concerne votre erreur, si je comprends bien, vous avez directement remplacés tous mes  $\theta_v$  des lignes de programmation par des  $x$  ?

Si c'est le cas, le conflit vient du fait que lorsque vous appelez la fonction depuis l'écran Home par (par exemple) `lagrange({1,2},{3,4},x)`, vous définissez déjà implicitement une variable locale  $x$  dans cette instruction, qui va être en conflit avec la variable générale  $x$  du programme qui va tenter de l'utiliser !!