## 1. Objectif :

On veut interpoler la fonction **f(x) = x/(x+sin(x)** à l'aide des polynômes d'interpolation de Newton et corriger le phénomène de Runge aux bords par la transformation de Tchebychev (polynômes de Tchebychev).

## 2. Interpolation de newton :

On utilise pour cela une fonction **newton(x,y)**, tels que x et y sont deux vecteurs de données correspondantes aux abscisses **x** et aux valeurs **y** de la fonction **y = f(x)** ou extraites de données expérimentales prises d'un essai au laboratoire.

D'après la méthode d'interpolation de Newton, on calcule tout d'abord les différences divisées pour former un tableau d'éléments triangulaire de dimension **nxn** :

Exemple pour n= 5 points :

| j=0 | j=1 | j=2 | j=3 | j=4 | j=5 |
|---|---|---|---|---|---|
| **x** | **y=f[$x_i$]** | **f[$x_i$,$x_{i+1}$]** | **f[$x_i$,$x_{i+1}$,$x_{i+2}$]** | **f[$x_i$,$x_{i+1}$,$x_{i+2}$,$x_{i+3}$]** | **f[$x_0$,$x_1$,$x_2$,$x_3$,$x_4$]** |
| x0 | y0 | * | * | * | * |
| x1 | y1 | f[$x_0$,$x_1$] | * | * | * |
| x2 | y2 | f[$x_1$,$x_2$] | f[$x_0$,$x_1$,$x_2$] | * | * |
| x3 | y3 | f[$x_2$,$x_3$] | f[$x_1$,$x_2$,$x_3$] | f[$x_0$,$x_1$,$x_2$,$x_3$] | * |
| X4 | y4 | f[$x_3$,$x_4$] | f[$x_7$,$x_8$,$x_9$] | f[$x_1$,$x_2$,$x_3$,$x_4$] | f[$x_0$,$x_1$,$x_2$,$x_3$,$x_4$] |

| j=0 | j=1 | j=2, i=1:4 | j=3, i=1:3 | j=4, i=1:2 | j=5, i=1 |
|---|---|---|---|---|---|
| **x** | **y=f[$x_i$]** | **f[$x_i$,$x_{i+1}$]** | **f[$x_i$,$x_{i+1}$,$x_{i+2}$]** | **f[$x_i$,$x_{i+1}$,$x_{i+2}$,$x_{i+3}$]** | **f[$x_0$,$x_1$,$x_2$,$x_3$,$x_4$]** |
| x0 | y0 | * | * | * | * |
| x1 | y1 | $\boldsymbol{y01} = \frac{y1-y0}{x1-x0}$ | * | * | * |
| x2 | y2 | $y12 = \frac{y2\ \ y1}{x2\ \ x1}$ | $\boldsymbol{y012} = \frac{y12-y01}{x2-x0}$ | * | * |
| x3 | y3 | $y23 = \frac{y3\ \ y2}{x3\ \ x2}$ | $y123 = \frac{y23\ \ y12}{x3\ \ x1}$ | $\boldsymbol{y0123} = \frac{y123-y012}{x3-x0}$ | * |
| X4 | y4 | $y34 = \frac{y4\ \ y3}{x4\ \ x3}$ | $y234 = \frac{y34\ \ y23}{x4\ \ x2}$ | $y1234 = \frac{y234\ \ y123}{x4\ \ x1}$ | $\boldsymbol{y01234} = \frac{y1234-y0123}{x4-x0}$ |

Polynôme d'interpolation de Newton :

$$p(x) = y0 + y01.(x-x0) + y012.(x-x0).(x-x1) + y0123.(x-x0).(x-x1).(x-x2) + y01234.(x-x0).(x-x1).(x-x2).(x-x3)$$

Qui peut-être écrit sous la forme récursive suivante :

$$p(x) = \left(\left(\left((\boldsymbol{y01234}.(x-x3) + \boldsymbol{y0123}).(x-x2) + \boldsymbol{y012}\right).(x-x1) + \boldsymbol{y01}\right).(x-x0) + \boldsymbol{y0}\right)$$

Qu'on peut l'exprimer par le code **scilab** :

```
p=c(n)
for k=n-1:-1:1
   p=p.*(x-x(k))+c(k)
end
```

Pour remplir le tableau, on utilise deux indices : **j=2,n** en colonne et **i=n-(j-1)** sur les lignes et pour chaque colonne. n peut être défini par **n=length(x).** Les éléments du tableau sont stockés dans la matrice c = y avant et après le calcul des différences divisées.

```
for k=2:n
    c(k:n)=(c(k:n)-c(k-1))./(x(k:n)-x(k-1))
end
```

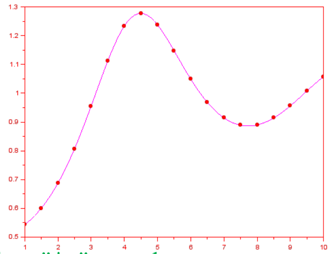
Pour former un polynôme avec **scilab**, on définit d'abord "**x**", remplacée par "**t**" dans la forme de newton, comme étant une variable non affectée (exclue de l'affectation) avant de récupérer ses coefficients en utilisant le code **scilab** suivant :

```
t=poly(0,'t')
p=c(n)
for k=n-1:-1:1
   p=p.*(t-x(k))+c(k)
end
```

Cette procédure force **scilab** à exprimer le résultat comme suivant :

$$p = 0.1509005 + 1.2735907t \quad 1.9017097t^1 + 1.6237301t^2 \quad 0.8115516t^3 + 0.2492765t^4 \quad 0.0454676t^5 + 0.0044545t^6 \quad 0.0001793t^7$$

Le tracé du graphique d'interpolation peut être fait en utilisant la fonction **scilab** d'évaluation **horner(p,x)** qui évalue le résultat du polynôme pour n'importe qu'elle valeur de **t**.



Le plus simple tracé du graphique est effectué par le code **scilab** suivant :

Exemples :

t=1:0.1:10, y=x./(x+sin(x)), plot(x, y) // pour tracer la fonction y = x./(x+sin(x)), plot(x,horner(p,x)) // pour le tracé du polynôme p(x). A changer x par t si 't' est la variable utilisée.

On peut utiliser optionnellement des marques de différents symboles et couleurs offerts par Scilab.

Exemple :

t=1:0.1:10, y=x./(x+sin(x)), plot(x, y, "*r") // * en rouge. Voir aussi "*b" pour le bleu, "*g" pour le vert.

On peut aussi utiliser la syntaxe suivante pour définir la couleur :

Exemple :

t=1:0.1:10, y=x./(x+sin(x)), plot(x, y, "color", "red") // Couleur "rouge" pour le tracé de la courbe.

## 3. Transformation de Tchebychev (Phénomène de Runge) :

La fonction d'interpolation de Newton présente dans les bornes de l'intervalle [a, b] des dispersions (oscillations) plus importantes que le degré du polynôme est élevé. Elle s'appelle le phénomène de Runge.

Polynôme de Tchebychev :

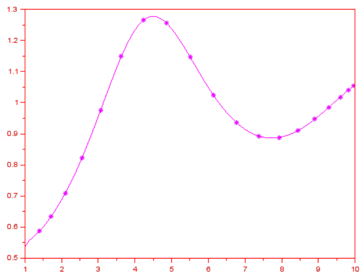
Le phénomène de Runge est dû en fait à l'équidistance des abscisses. Pour cette raison, on se sert aux polynômes de Tchebychev pour corriger ces fluctuations aux bords. Ces polynômes permettent de rendre les données des abscisses serrées aux bords et élargies à l'intérieur. Pour cela une transformation des abscisses est possible à l'aide de la formule suivante :

$$x_k = \frac{1}{2}(a+b) + \frac{1}{2}(b-a).\cos\left(\frac{2k-1}{2n}.\pi\right), \qquad k = \overline{1,n}$$

Avec des données en abscisses non-équidistantes, l'application de l'interpolation de Newton de nouveau diminuera nettement le phénomène de Runge aux bords.

Le code de transformation Tchebychev de **scilab** est assez simple (n : nombre de points des données d'interpolation) dans l'intervalle [a,b] :

```
function [ap]=chebyshev(a, b)
  n=(b-a)+1
    x=(a+b)/2+(b-a)/2*cos((2*(n-(1:0.5:n))+1)*%pi/(2*n+2));
    y=x./(x+sin(x))
    ap =newton(x,y);
  clf()
  plot(t,horner(ap,t))
  plot(x,y,"*g")
endfunction
```



## 4. Détails du travail demandé :

1. Introduire les données expérimentales ou extraites de la fonction réelle **f(x) = x/(x+sin(x))** présentées sous forme de deux vecteurs **x** et **y** à **n** dimensions ;
2. Confection d'une fonction d'interpolation de Newton qui permet de :
   - Calculer les différences divisées ;
   - Former et d'exprimer (afficher) le polynôme d'interpolation de Newton ;
3. Evaluer le polynôme d'interpolation de Newton par l'utilisation de la fonction prédéfinie de **scilab horner (p,x)** et faire son tracé sur un graphique. Une comparaison avec le tracé de la courbe de la fonction réelle est nécessaire ;
4. Confection d'une fonction de transformation de Tchebychev qui permet de :
   - Transformer les abscisses **x** pour les rendre non-équidistantes et calculer de nouveau les valeurs **y** ;
   - Appeler la fonction de Newton de nouveau pour refaire l'interpolation de Newton avec les nouveau vecteurs **x** et **y** (déterminés de la transformation de Tchebychev) et afficher le polynôme d'interpolation de Newton ;
   - Tracer la courbe d'interpolation sur le même graphique précédent ainsi que les points de Tchebychev par des marques.
5. Confectionner un programme qui permet :
   - d'introduire les différents paramètres de raffinement des résultats théoriques ou expérimentaux : **a**, **b**, **n** (nombre de points pour élever le degré du polynôme), **d** (le pas des abscisses pour améliorer le tracé d'une courbe),
   - d'utiliser les deux fonctions, de Newton et de Tchebychev pour le calcul, l'affichage des différents résultats et le tracé de différents graphiques demandés.
6. Conclure.