T.P. numéro V

Précision de calcul. Codage des flottants.

Rappels de cours

Flottant simple precision (float) : Sur un PC le type `float` représente les nombres réels simple précision. Ils sont codés sur 32 bits. Leur format d'écriture est `%f`.

Flottant double precision (double) : Sur un PC le type `double` représente les nombres réels double précision. Ils sont codés sur 64 bits. Leur format d'écriture est `%lf`

1 Résolution d'une équation du 2nd degré

On considère une fonction $f(x) = ax^2 + bx + c$. Les solutions de l'équation $f(x) = 0$ sont données par :

$$x_{\pm} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{1}$$

Exercice V.1

1. Ecrire un algorithme qui calcule et affiche à l'écran les solutions `sol1` et `sol2` de l'équation $f(x) = 0$ avec $a = 1$, $b = 4 \times 10^5$ et $c = 3$. On utilisera des variables de type **float** .
   Pour l'utilisation des fonctions mathématiques (puissance et racine carrée), on utilisera la librairie <math.h>. La fonction puissance m^n s'écrit pow(m,n) et la fonction racine carrée $\sqrt{m}$ s'écrit sqrt (m).
   Le programme sera compilé avec l'option `-lm`
2. Vérifier si les solutions obtenues sont correctes. Pour cela on affichera à l'écran les résultats des calculs `a*sol1^2+b*sol1+c` et `a*sol2+b*sol2+c`. Que se passe t'il ?
3. Pourquoi les résultats obtenus ne sont pas corrects ?
4. Améliorer la précision du calcul.
5. Conclusion.

---

2 Calcul d'une intégrale

On se propose de calculer l'intégrale

$$I = \int_a^b f(x)\, \mathrm{d}x \tag{2}$$

par la méthode composite des trapèzes. Cette méthode consiste à diviser le domaine d'intégration $[a, b]$ en N intervalles $[x_0, x_1]$, $\ldots$, $[x_{i-1}, x_i]$, $\ldots$, $[x_{N-1}, x_N]$ tous de longueur identique $h = (b - a)/N$. On a donc $x_0 = a$, $x_1 = a+h$, $\ldots$, $x_i = a + ih$, $\ldots$, $x_N = b$ (voir Fig. 1).
Dans chaque intervalle $[x_{i-1}\,,\ x_i]$ l'intégrale, c'est-à-dire l'aire située sous la courbe, est approximée par l'aire du trapèze

$$\frac{h}{2}\,[f(x_{i-1}) + f(x_i)] \tag{3}$$

L'intégrale complète sur $[a\,,\ b]$ est alors évaluée en additionnant toutes les aires de ces trapèzes et est donc approximée par l'expression :

$$I \approx \frac{h}{2}\,[f(x_0) + f(x_N)] + h\sum_{i=1}^{N-1} f(x_i) \tag{4}$$

On montre de plus que l'erreur relative sur la valeur de l'intégrale commise en utilisant cette méthode est :

$$E_T \propto Nh^3 \tag{5}$$

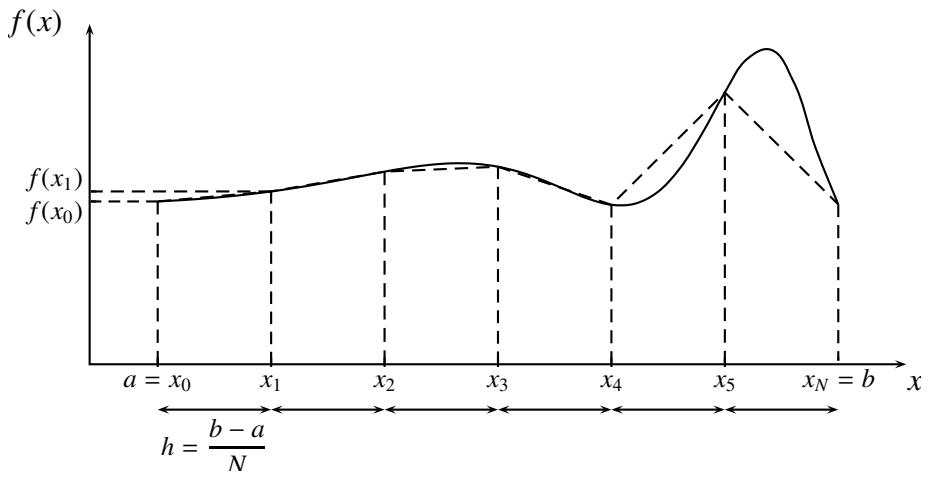

Fig. 1 : Exemple pour $N = 5$. L'intégrale de $f(x)$ entre a et b est estimée par la somme des aires des N trapèzes représentés en pointillé.

Nous allons étudier la convergence du résultat de l'intégration numérique de la fonction $f(x) = \sin(x^2)$ en fonction du nombre d'intervalles N.
Sauvegarder et éditer le programme integrale.c. Ce programme (inachevé) est organisé de la façon suivante :
– Un programme principal.
– Une fonction f () dans laquelle est évaluée la fonction à intégrer.
– Une fonction `trapezes()` qui calcule l'intégrale de $f(x)$ par la méthode des trapèzes.
La programmation des fonctions f () et trapezes () fera l'objet de l'exercice V.3.

Exercice V.2

Afin de mieux comprendre le programme integrale.c , répondre aux questions suivantes :

1. Quelles sont les données demandées à l'utilisateur dans le programme et leur signification ?
2. A quoi sert la boucle **for** ?
3. Quelle est l'intérêt de calculer l'intégrale avec 1000 intervalles ?
4. Que fait la ligne
   fprintf (fichier , "%d %f %f \n",N, erreur , Integrale_N); ?
5. Quel est le nom du fichier dans lequel les résultats sont écrits ?
6. Comment est calculée l'erreur ?

Exercice V.3

1. Compléter la définition des fonctions f () qui renvoie $f(x)$ et trapezes () qui renvoie la valeur approchée de l'intégrale. Faire afficher la valeur « exacte » de l'intégrale par le programme principal.
2. Compiler (avec l'option `-lm`) et exécuter le programme afin de calculer l'intégrale de la fonction $f(x)$ entre 0 et 1 pour un nombre d'intervalles variant de 2 à $N_{\max} = 30$. Quelle est la valeur « exacte » de l'intégrale ? Quelle est l'erreur relative commise avec $N = 30$?
3. Tracer en échelle log-log la variation de l'erreur en fonction de N. Comment varie l'erreur avec N ?
   On utilisera la commande `xmgrace -log xy trap.dat`
4. En échelle log-log, une fonction du type $y = ax^b$ devient une droite d'équation $\ln y = \ln a + b \ln x$. En supposant que l'erreur varie comme $E = aN^b$, déterminer une valeur approchée de b à l'aide des valeurs contenues dans le fichier `trap.dat`. Quelle devrait être théoriquement la valeur de b compte tenu de l'équation (5) pour l'erreur ? Comparer.
5. En prenant $N_{\max} = 700$, étudier comment varie l'erreur en fonction de N pour N grand (on prendra une échelle de 10^{-8} et 10^{-1} sur l'axe des ordonnées). Que se passe t'il ?
6. Modifier le programme pour remedier à ce problème et observer la différence.

Remarque : En calculant cette intégrale par une autre méthode plus précise, on obtient :

$$I = 0.31026830172338110181$$

Listing des codes à télécharger

Code integrale.c

```
#include <stdio.h>
#include <math.h>


/* Interface pour les fonctions appelees */
float f (float x );
float trapezes (float a, float b, int N);

/****** Programme principal *********/
int main (void) {
  int  Nmax, N;
  float a, b, erreur, Integrale_Exacte, Integrale_N;
  FILE *fichier;
  char Nom_du_fichier[80]= "trap.dat";

  /*****************************************/
  /* Saisit les parametres                 */
  /*****************************************/
  printf("Entrer la borne inferieure a: ");
  scanf("%f",&a);
  printf("Entrer la borne supérieure b: ");
  scanf("%f",&b);
  printf("Entrer le nombre d'intervalles max Nmax: ");
  scanf("%d",&Nmax);

  /*****************************************/
  /* Calcul de l'integrale pour chaque N et  */
  /* écriture dans un fichier              */
  /*****************************************/
  /* ouvre le fichier de sortie */
  fichier = fopen(Nom_du_fichier,"w");
  if (fichier == NULL)
    {
      printf ("Le fichier %s n'a pas pu etre ouvert\n",Nom_du_fichier);
      return;
    }

/**** Calcule la valeur exacte de l'integrale (N grand) ****/
  Integrale_Exacte = trapezes(a,b,1000);

/**** Calcule la valeur de l'integrale et son erreur en fonction de N ****/
  for (N = 2; N <= Nmax ; N+=1) {
      Integrale_N = trapezes(a,b,N);
      erreur = fabs((Integrale_N - Integrale_Exacte)/Integrale_Exacte);
      fprintf(fichier,"%3d %.10f %.10f\n",N,erreur,Integrale_N);
  }
  fclose(fichier);
  return;
}

/*******************************************************/
/* Fonctions                                           */
/*******************************************************/
float f (float x) {
}

/*             ---------------------------             */
float trapezes (float a, float b, int N) {
}
```