计　算　方　法

实验二 Romberg 积分法

姓名　　郭茁宁

学号　　1183710109

院系　　计算机科学与技术学院

专业　　软件工程

哈尔滨工业大学

实验报告二

1.题目（摘要）

利用龙贝格(Romberg)积分法计算积分$\int_a^b f(x)dx$

输入：a, b, N, ε
输出： 龙贝格$T-$数表

2.前言（目的和意义）

目的：利用龙贝格(Romberg)积分法计算积分$\int_a^b f(x)dx$

意义：学习根据实际问题建立的数学模型，针对数学模型的特点确定适当的计算方法，编制出计算机能够执行的计算程序，输入计算机，进行调试，完成运算等数值计算的过程。不只会套用教科书中的标准程序进行数值计算，独立地将学过的数值算法编制成计算机程序，灵活应用已经掌握的算法求解综合性较大的课题。理解数值计算程序结构化的思想，提高编程能力，加深对“计算方法”课程内容的理解和掌握，进一步奠定从事数值计算工作的基础。具体可以利用所掌握的“高级语言”顺利地编制出计算机程序，上机实习，完成实验环节的教学要求。不简单地套用现成的标准程序完成实验题目，把重点放在对算法的理解、程序的优化设计、上机调试和计算结果分析上，达到实验课的目的。

3.数学原理

利用复化梯形求积公式、复化辛普生求积公式、复化柯特斯求积公式的误差估计式计算积分$\int_a^b f(x)dx$。记$h=\frac{b-a}{n}$，$x_k=a+k\cdot h$，$k=0,1,\cdots,n$，其计算公式：

$$T_n=\frac{1}{2}h\sum_{k=1}^{n}[f(x_{k-1})+f(x_k)]$$

$$T_{2n}=\frac{1}{2}T_n+\frac{1}{2}h\sum_{k=1}^{n}f(x_k-\frac{1}{2}h)$$

$$S_n=\frac{1}{3}(4T_{2n}-T_n)$$

$$C_n=\frac{1}{15}(16S_{2n}-S_n)$$

$$R_n=\frac{1}{63}(64C_{2n}-C_n)$$

一般地，利用龙贝格算法计算积分，要输出所谓的$T-$数表

$$\begin{array}{lllll} T_1 & & & & \\ T_2 & S_1 & & & \\ T_4 & S_2 & C_1 & & \\ T_8 & S_4 & C_2 & R_1 & \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{array}$$

4.程序设计流程

1．准备初值，计算

$$T_{0,0}=\frac{a-b}{2}[f(a)+f(b)]$$

且 $k \leftarrow 0$（k 为等份次数）

2．按梯形公式的递推关系，计算

$$T_{0,k+1}=\frac{1}{2}T_{0,k}+\frac{b-a}{2^{k+1}}\sum_{i=0}^{2^k-1}f(a+\frac{b-a}{2^k}(i+\frac{1}{2}))$$

3．按龙贝格公式计算加速值

$$T_{0,k-m}\leftarrow T_{m,k-m}=\frac{4^m T_{m-1,k+1-m}-T_{m-1,k-m}}{4^m-1} \qquad m=0,1,2,\cdots,k$$

4．精度控制。对给定的精度 ε，若

$$|T_{m,0}-T_{m-1,0}|<\varepsilon$$

则终止计算，并取 $T_{0,s}\leftarrow T_{m,s}$ 作为所求结果；否则 $k\leftarrow k+1$，重复 2~4 步，直到满足精度为止。

核心代码:

```
#include <cmath>
#include <cstdio>
#include <iostream>
using namespace std;
#define N 100

int n;
long double a, b, e, T[N][N] = {{0.0}};

long double f(long double x) { return x * x * exp(x); }
long double x(int i, long double h) { return a + h * i; }

int main() {
    scanf("%llf%llf%llf%d", &a, &b, &e, &n);
    int k = 0;
    for (; k < n; k++) {
        long double h = (b - a) / pow(2, k), sum = 0.0;
        for (int i = 1; i <= pow(2, k) - 1; i++) sum += f(x(i,
 h));
        T[k][0] = 0.5 * h * (f(a) + 2 * sum + f(b));
        for (int m = 1; m <= k; m++)
```

```
            T[k][m] =
                (pow(4, m) * T[k][m - 1] - T[k - 1][m - 1]) /
(pow(4, m) - 1);
        if (k > 0) {
            if (fabs(T[k][0] - T[k][k]) <= e) break;
            cout << T[k][0] << '\t' << T[k][k] << endl;
        }
    }
    cout << T[k - 1][0] << '\n' << T[k - 1][k - 1] << endl;
    return 0;
}
```

5.实验结果、结论与讨论

1. $\int_0^1 x^2 e^x \, dx$， $\varepsilon = 10^{-6}$

1.3591409										
0.8856606	0.7278338									
0.7605963	0.7189082	0.7183132								
0.7288902	0.7183215	0.7182823	0.7182819							
0.7209358	0.7182843	0.7182818	0.7182818	0.7182818						
0.7189454	0.7182820	0.7182818	0.7182818	0.7182818	0.7182818					
0.7184477	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818				
0.7183233	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818			
0.7182922	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818		
0.7182844	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	
0.7182825	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818	0.7182818

$$\therefore \int_0^1 x^2 e^x \, dx \approx 0.71828$$

2. $\int_1^3 e^x \sin x \, dx$， $\varepsilon = 10^{-6}$

5.1218264												
9.2797629	10.6657417											
10.5205543	10.9341514	10.9520454										
10.8420435	10.9492065	10.9502102	10.9501811									
10.9230939	10.9501107	10.9501710	10.9501704	10.9501703								
10.9433984	10.9501666	10.9501703	10.9501703	10.9501703	10.9501703							
10.9484772	10.9501701	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703						
10.9497470	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703					
10.9500645	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703				
10.9501439	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703			
10.9501637	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703		
10.9501687	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	
10.9501699	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.9501703	10.950170

$$\therefore \int_1^3 e^x \sin x \, dx \approx 10.95017$$

3. $\int_0^1 \frac{4}{1+x^2} dx$， $\varepsilon = 10^{-6}$

3.0000000			
3.1000000	3.1333333		
3.1311765	3.1415686	3.1421176	
3.1389885	3.1415925	3.1415941	3.1415858

3.1409416	3.1415927	3.1415927	3.1415926	3.1415927					
3.1414299	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927				
3.1415520	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927			
3.1415825	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927		
3.1415901	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	
3.1415920	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927	3.1415927

$$\therefore \int_0^1 \frac{4}{1+x^2}dx \approx 3.14159$$

4. $\int_0^1 \frac{1}{x+1}dx$，$\varepsilon = 10^{-6}$

0.7500000								
0.7083333	0.6944444							
0.6970238	0.6932540	0.6931746						
0.6941219	0.6931545	0.6931479	0.6931475					
0.6933912	0.6931477	0.6931472	0.6931472	0.6931472				
0.6932082	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472			
0.6931624	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472		
0.6931510	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	
0.6931481	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472	0.6931472

$$\therefore \int_0^1 \frac{1}{x+1}dx \approx 0.69315$$

思考题：输入的参数$\boldsymbol{N}$有什么意义？

输入的参数N越大，在有限区间上分段越小，由复化梯形公式的误差得，计算精度越高。