**实验报告**

计51 柴华君 2015011377

实验一：误差

**1. 实验要求**

试用单精度 float 计算xn， 问 n 为何值时能满足精度要求？令ln2的准确值为0.693147190546

**2. 算法描述**

while循环计算ln2的累加式，当误差小于规定值时输出对应的n值

1. **int** main(){
2. **double** ln2 = 0.693147190546;
3. **float** epsilon = 0.00005;
4. float xn = 1;
5. **int** k = 1;  //实际指数的k-1 为奇偶之分
6. **while**(fabs(xn - ln2) >= epsilon){
7. k++;
8. **if**(k % 2 == 0)  //k为偶
9. xn += (-1 \* 1.0) / k;
10. **else**
11. xn += 1.0 / k;
12. }
13. cout << " n is " << k;
15. **return** 0;
16. }

**3. 程序清单**

Exp1.cpp Exp1.exe

**4. 运行结果**

­­­ n为8975，与实际的n值19999差距较大，主要是因为float存在截断误差，而且每次计算会使误差不断累加，所以导致最终的结果与实际值差距较大

**5. 体会和展望**

误差实验整体比较简单，但让我更直观地认识到了截断误差对最终结果可能带来的巨大影响。

实验二：插值法

**1. 实验要求**

对[-5,5]作等距划分,并对Runge给出的函数

作 Lagrange 插值和三次样条插值，观察 Runge 现象并思考改进策略

**2. 算法描述**

Lagrange代数插值，用书上给出的公式进行实现。

1. **float** Ln\_x(**float** x[], **float** y[], **int** num, **float** \_x){  //lagrange插值
2. **float** Ln\_x = 0;
3. **float** wn\_1\_ans = wn\_1(num, x, \_x);
4. **for**(**int** i = 0; i <= num; i++){
5. **if**(\_x - x[i] == 0){
6. Ln\_x = y[i];
7. **break**;
8. }
9. Ln\_x += y[i] \* wn\_1\_ans / ((\_x - x[i]) \* \_wn\_1(i, num, x));
10. }
11. **return** Ln\_x;
12. }

三次样条插值利用课本6.8的公式进行实现，参考了例7的解题思路。对于M的求解，使用了ppt提及的追赶法。

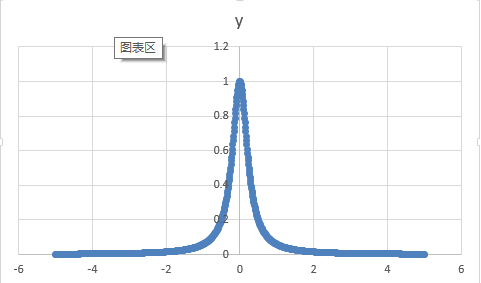
1. **float** S(**float** x[], **float** y[], **float** m[], **float** h, **int** j, **float** \_x){
2. **float** part1, part2, part3, part4;
3. part1 = m[j] \* (x[j+1] - \_x) \* (x[j+1] - \_x) \* (x[j+1] - \_x) / (6 \* h);
4. part2 = m[j + 1] \* (\_x - x[j]) \* (\_x - x[j]) \* (\_x - x[j]) / (6 \* h);
5. part3 = (y[j] - m[j] \* h \* h / 6) \* (x[j + 1] - \_x) / h;
6. part4 = (y[j + 1] - m[j+1] \* h \* h / 6) \* (\_x - x[j]) / h;
7. **return** part1 + part2 + part3 + part4;
8. }

**3. 程序清单**

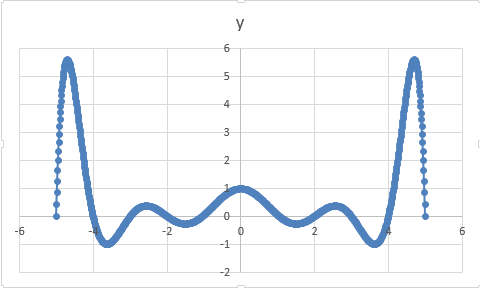
Exp2.cpp Exp2.exe

**4. 运行结果**

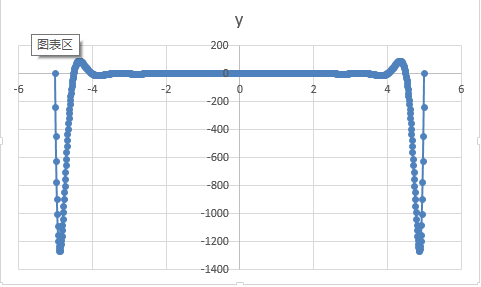
­­­ 原函数图像：



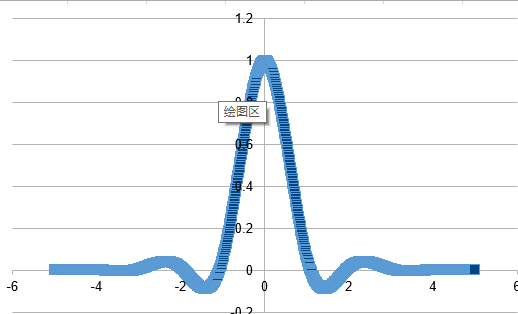
L10(x)图像：



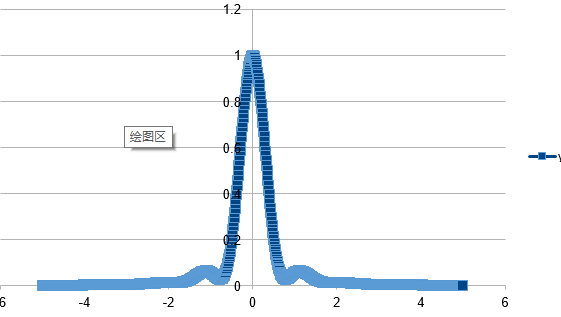
L20(x)图像：



S10(x)图像：



S20(x)图像：



3> 不同：lagrange插值与原图像误差相对较大，且丢失了原函数的凹凸性，L10(x)的误差大于L20(x)的误差

三次样条插值结果与原图像误差较小，且较好地保持了原图像的凹凸性，S10(x)的误差同样大于S20(x)的误差

4> x=4.8处的误差 f(4.8) = 0.0027053 S20(x) = 0.0027007 L20(x) = 0.0027050

在x=4.8处，lagrange插值的误差更小，可能是由于数据点多，lagrange插值在插值点附近收敛较快造成的。

**5. 体会和展望**

通过实验，对于lagrange插值和三次样条插值的函数求解过程更加清楚，且更直观地看到了两者在求解时的优劣。对于我更好地理解两个插值函数，有着很大的帮助。