实验一 截断误差与舍入误差

计21班 杨俊 2012011400

一、实验内容：

已知$\ln 2 = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots + (-1)^{n-1}\frac{1}{n} + \cdots$，

记$x_n = \sum_{k=1}^{n} \frac{(-1)^{k-1}}{k}$，

则x_n构成逼近$\ln 2$的数列。根据交错级数知识，有估计式$|x_n - \ln 2| < \frac{1}{n+1}$。

记$|x_n - \ln 2| < \varepsilon$，

1、若取ε_1=1/2×10^(-5)，试用单精度计算x_n，问n为何值时能满足精度要求？

2、又，若取ε_2=12×10^(−5)时，n为何值时能满足精度要求？对ε_1和ε_2，理论上的n值与实际计算的n值有何不同？为什么？（取十二位有效数字作为“准确值”：ln2=0.693147190546。）

二、解题思路及算法设计

单精度数用浮点数float实现，双精度数用double实现。当误差大于约定值时，则对n进行继续运算，而当误差小于这个值时返回当前的n,并且退出递归。

三、实验结果及结论


```
C:\Users\yangjun\Desktop\数值分析\homework1_2012011400_杨俊>a
when the E is 0.000005:  n=40872
when the E is 0.0000005: n=61344
```

当误差为0.000005时，n=40872

当误差为0.0000005时，n=61344

而理论上，若误差取0.000005，n应该为1/0.000005-1=199999;

若误差取0.0000005，n应该为1/0.0000005-1=1999999；明显是理论上面的n比较大，而实际得到的n比较小。我认为这是有两方面的原因引起的，第一方面，理论上用的公式$\varepsilon=1/(n+1)$是一个比较宽的上界，并不能精确地估计实际的大小；另外一方面，由于在进行加法计算时，用的是float而不是double，而float表示的单精度浮点数有效数字为7位，这使得用float会存在一定的舍入误差，使得当后来n比较大的时候而$1/n$变得非常小，导致了大数吃小数的情况，使得误差积累。

通过本次编程实践，得到不同的数据类型的误差是不同的，因为有效位数的不同，数据对舍入误差比较敏感，而如果出现大数和小数相加的情况，那么就会使小数后几位有效数字被忽略从而导致误差。类似的情况可以通过将近大小的数进行相加，先加比较小的数后加比较大的数，这样可以使误差减小；还有另外一种方法是用双精度浮点数，这种方法固然有效数值也比较多，但是导致的是内存占用比较多，所以根据具体情况进行取舍。