# 计算方法第一次上机作业

曾梦辰*

2024 年 5 月 8 日

**摘要**

本次作业完成教材第 20 页 17 题, 分析舍入误差与有效数.

## 1 问题

设 $S_N = \sum_{j=2}^{N} \frac{1}{j^2-1}$, 其精确值为 $\frac{1}{2}\left(\frac{3}{2} - \frac{1}{N} - \frac{1}{N+1}\right)$.

(1) 编写按从大到小 (求和指标上升) 的顺序计算 $S_N$ 的通用程序, 即

$$S_N = \frac{1}{2^2-1} + \frac{1}{3^2-1} + \cdots + \frac{1}{N^2-1}$$

(2) 编写按从小到大 (求和指标下降) 的顺序计算 $S_N$ 的通用程序, 即

$$S_N = \frac{1}{N^2-1} + \frac{1}{(N-1)^2-1} + \cdots + \frac{1}{2^2-1}$$

(3) 按以上程序计算 $S_{10^2}, S_{10^4}, S_{10^6}$, 并指出有效位数, 数据使用单精度格式储存. 分析以上结果.

## 2 算法

我们分别使用以下算法进行计算.

*学号: 202011999050

**Algorithm 1:** 从大到小相加

**Input:** Natural number $N$

**Output:** $\sum_{2\leq j\leq N} \frac{1}{j^2-1}$

Define $j = 2, S = 0$;

**for** $j \leq N$ **do**

  $S+ = \frac{1}{j^2-1}$;

  $j+ = 1$

**end**

return $S$

**Algorithm 2:** 从小到大相加

**Input:** Natural number $N$

**Output:** $\sum_{2\leq j\leq N} \frac{1}{j^2-1}$

Define $j = N, S = 0$;

**for** $j \geq 2$ **do**

  $S+ = \frac{1}{j^2-1}$;

  $j- = 1$

**end**

return $S$

# 3 运行结果

程序源代码参考文件夹中的`program.c`, 以下列举运行结果.

| $N$ | 精确值 | 从大到小 | 从小到大 |
|---|---|---|---|
| $10^2$ | 0.74004950 | 0.74004948 | 0.74004954 |
| $10^4$ | 0.74990000 | 0.74985212 | 0.74989998 |
| $10^6$ | 0.74990000 | 0.74985212 | 0.74999899 |

表 1: 运行结果

使用 $\varepsilon_1(N)$ 表示从大到小计算时的绝对误差限, $\varepsilon_2(N)$ 表示从小到大计算时的绝对误差限. 当 $N = 10^2$ 时, 从大到小计算有 7 位有效数, 从小到大计算有 7 位有效数, 且 $\varepsilon_1(10^2) < \varepsilon_2(10^2)$; 当 $N = 10^4$ 时, 从大到小计算有 3 位有效数, 从小到大计算有 3 位有效数, 且 $\varepsilon_1(10^4) > \varepsilon_2(10^4)$; 当 $N = 10^6$ 时, 从大到小计算有 3 为有效数, 从小到大计算有 4 位有效数, 自然 $\varepsilon_1(10^6) > \varepsilon_2(10^6)$.

# 4 结果分析

通过以上结果可以得到两点:

1. 随着 $N$ 增大, 两种算法的有效位数均在下降. 这是因为随着 $N$ 增大, 计算量增大, 舍入误差逐渐累积导致绝对误差限增大.
2. 从大到小计算的有效数位数总是不大于从小到大计算的有效数位数. 这是因为计算时, 为了执行大数加小数的加法, 计算机会对齐大数与小数的有效数, 导致大数加小数时在计算后期大量损失有效数, 从而产生更大的误差.

# 5 运行环境

程序运行环境: Microsoft Corporation Surface Pro 6, Windows 11 23H2.
编译器: gcc version 13.2.0 (Rev3, Built by MSYS2 project).