

面向对象程序设计基础作业二 设计文档

1. 模型部分

a. 功能简述

本程序实现了如下功能：在系统提示 “Input two numbers:” 后，输入两个正整数，之后会输出其最大公约数、最小公倍数及计算用时。

b. 数据结构

```
#ifndef HW2_CPP_INT_H
#define HW2_CPP_INT_H
class intpair
{
public:
    intpair();
    ~intpair();
private:
    int *value1, *value2;
    long long calc_gcd();
    long long *gcd;
    long long calc_lcm();
    long long *lcm;
    long long *timeused;
public:
    void output();
};
#endif //HW2_CPP_INT_H
```

本程序包括主函数源文件（main.cpp）和 1 个额外源文件及其头文件（CPP_int.cpp/h）。在 CPP_int.h 中定义了类 intpair，暂时包括一个存放两个 int 整数值的私有成员变量和部分函数接口，并在对应源文件中进行了实现。

其中，构造函数 intpair::intpair() 包括了申请内存、输入、输出、计算最大公约数 calc_gcd()、计算最小公倍数 calc_lcm()、记录所用时间的全部工作，因此调用的参数及函数均隔离为私有参数及私有函数；output() 函数用于输出，是一个对外的接口，因此设置为公有函数。

c. 算法

在最大公约数上，本程序使用辗转相除法，其数学原理为：

$$\gcd(a, b) = \gcd(a, a \bmod b)$$

具体而言，对于输入的数据，我们反复用大数对小数取模替换小数，直到有一方为 0 时，另一方就是所求的最大公约数。这个过程的时间复杂度是 $o(\log_2 n)$ 的。

在最小公倍数上，本程序直接运用如下公式：

$$\text{lcm}(a, b) = \frac{a \times b}{\gcd(a, b)}$$

这个过程的时间复杂度是 $o(1)$ 的。

在计算时间上，由于本程序总体的时间复杂度很低，在现代计算机上运行时间非常短。因此，本程序每次运行时，会重复计算 1 000 000 次最大公约数与最小公倍数，平均后折算成纳秒输出。

2. 验证部分：时间复杂度验证

由于具体的运行时常与选择的数字非线性地密切相关，本文仅提供若干数据点及其对应用时。（在一台 MacBook Pro 2020 上测试）

value1	value2	用时 (ns)
13	15	44
287	193	64
4043	2357	82

10562	33725	81
213598	426389	71

可以看出，用时与数据规模并非单调相关，但可以基本认为用时数量级为 10^{-9} s。考虑时间复杂度 $\log_2 n$ ， n 取 10^6 时约 20（次计算），结合普遍认为个人计算机每秒计算次数大约 10^8 - 10^9 次，基本在数量级上吻合。