面向对象程序设计基础作业一 设计文档

1. 模型部分
   1. 功能简述

本程序实现了如下功能：从控制台输入任一字符串，程序将进行合法性判断，若为一整数则输出小于等于该整数的所有正偶数之和。该和采用两种不同方法计算（见1.b）。

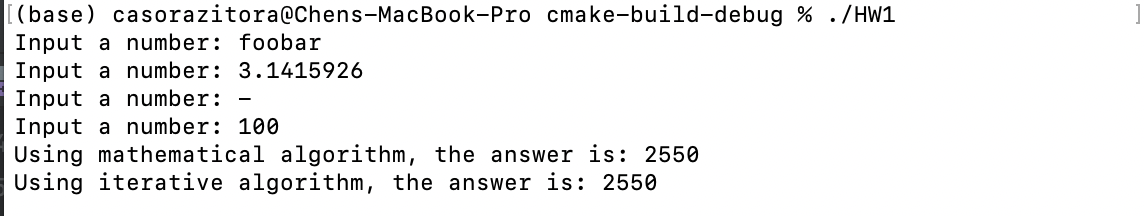
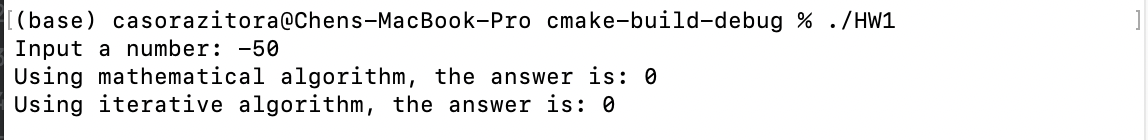
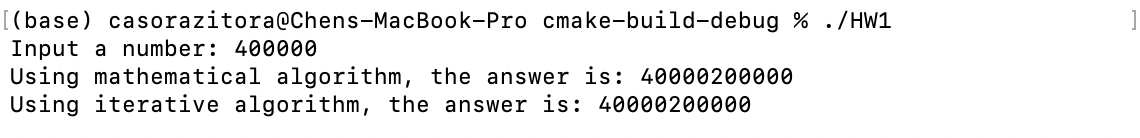


图 1 使用示例（从上至下分别展示正整数输入、负整数输入、非法数据输入）

同时，本程序自带交叉验证功能。从命令行执行本程序时，可以通过在指令后添加：**-t [正整数]**，程序将验证小于等于该正整数的整数是否计算正确（见2.）。

* 1. 算法

本程序采用两种方法计算所期望的和，称为“公式法”和“循环法”。本部分的讨论过程假设输入的数是合法的正数n。设：（在程序中这一步由temp = n >> 2实现）。

公式法，即利用数学公式：

在本题中，。计算此公式即得结果。本法的时间复杂度为O(1)。

循环法，即利用定义式：

利用循环累加模拟此过程即得结果。本法的时间复杂度为O(n)。

从时间复杂度上考虑，公式法相比循环法的优势是明显的，尤其在n较大（>1e8）时，公式法的计算时间几乎可以忽略不计，而循环法的计算时间变得肉眼可见。但由于循环法的本质是模拟法，模拟的过程相比公式在一些层次上更加可靠。

* 1. 数据结构

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\_getint.cpp/h）。在CPP\_getint.h中定义了类customdata::cpp\_int，暂时包括一个存放整数值的私有成员变量和部分函数接口（见图 2），并在对应源文件中进行了实现。

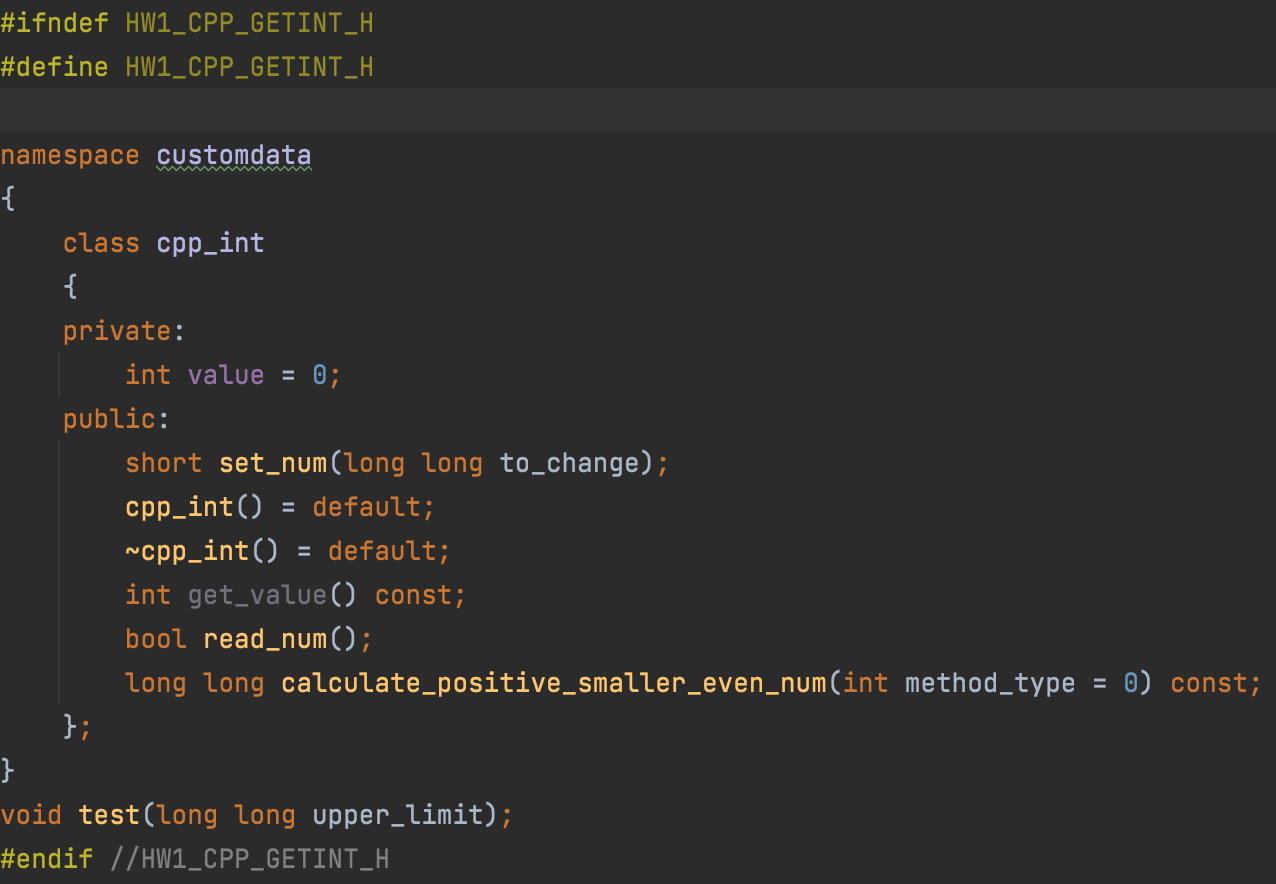


图 2 CPP\_getint.h

1. 验证部分

本程序从命令行执行时可使用自动验证程序。

从控制台调用此程序时，若读入了-t [正整数]指令，程序将进行交叉比对。比对的过程是：从0开始，计算两种方法得到的结果并比较是否一致，若一致则以1073741824为初始步长检查下一数字；若不一致，或者此时将要验证的数字已经大于读入正整数，则回退一个步长并减半步长，直到步长为0，就输出验证到的最大数字。这一验证过程将调用约数量级的计算过程，每个计算过程的时间复杂度均为O(n)，因此整体的验证时间复杂度大约为O(nlogn)。

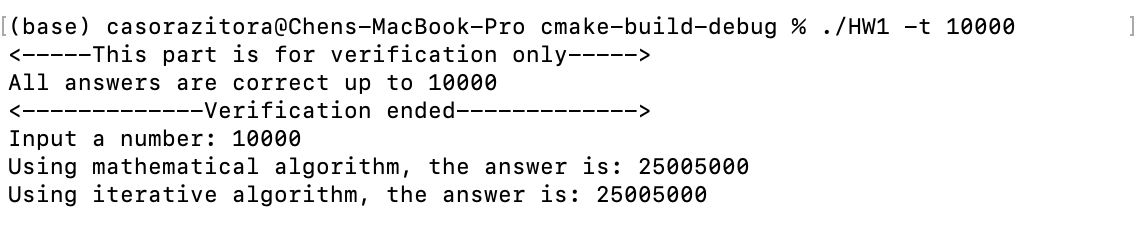


图 3 验证过程示例

这一验证方法并非逐一验证：我们假设，越大的数字越容易出现错误（最常见的是由于数据规模溢出），且如果验证了A与B均正确，则假设A、B间的数都正确。同时，由于我们只验证了两种算法是否一致，无法完全排除算法本身的瑕疵。