

中山大学数据科学与计算机学院本科生实验报告

(2016 学年春季学期)

课程名称: Algorithm design

任课教师: 张子臻

年级	1501	专业(方向)	移动信息工程
学号	15352015	姓名	曹广杰
电话	13727022190	Email	1553118845@qq.com
开始日期	2017/5/21	完成日期	2017/5/21

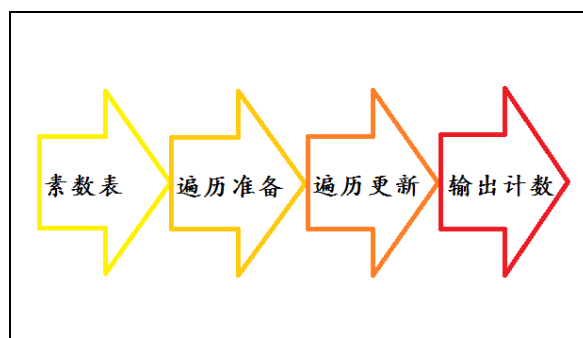
1. 实验题目



2. 实验目的

- 1) 给出一个正整数, 求出它有多少种方法可以表示成连续的素数的和。
- 2) 例如 $53 = 5 + 7 + 11 + 13 + 17 = 53$, 共有两种方法。;

3. 程序设计



具体解释如下:

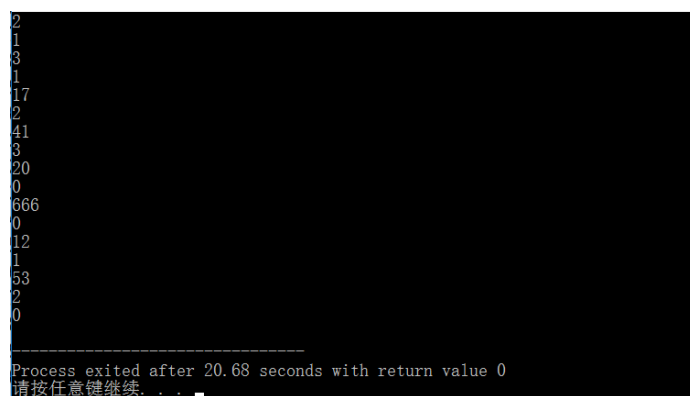
1. 首先, 输入的数据需要储存。这里使用数组储存质数表, 之后遍历调用。此处可以有多种方式进行储存, 但是由于储存的数据与其他的数据之间的关联少, 产生的结构简单即可应用, 这里就不再讨论该问题。

2. 输入数据储存之后，开始进行计算，质数表的所有起讫点之间的和：

- a. 素数判断。检查从 2 到 10000 的素数，对所有小于等于被检查数 n 的开方的数字进行检查，看是否为 n 的因子。
- b. 遍历的方案确定。由于题意要求检查所有素数的顺序组合是否符合要求，就免不了异常声势浩大的遍历。但是只要是遍历，时间复杂度都会非常高，需谨慎对待。这里笔者需要做点遍历之前的准备。
- c. 遍历的准备。由于笔者的遍历要求计算任意起讫点之间的数据和，计算量非常大，笔者需要做一些准备。在笔者完成了关于所有的循环的实现之后，发现了 3 重循环嵌套内部求和过程有大量的重复运算，这里简直是计算的毒瘤。于是笔者将其放在 2 重循环之外，使用原有的素数数组进行储存。
- d. 遍历与更新计数器。由之前的准备工作，可以得到一个计算复杂度为平方的算法。在此算法中，所有的信息都会被检查一遍，并且每一次有一个符合要求的黎曼和，就会更新局部计数器。

3. 输出计数器；

4. 程序运行与测试



```
2
1
3
1
17
2
41
3
20
0
666
0
12
1
53
2
0
-----
Process exited after 20.68 seconds with return value 0
请按任意键继续. . .
```

5. 实验总结与心得

1. 本次实验中需要强调的位置是对于重复计算的处理方式。

重复计算的处理，如果直接使用三重循环进行计算，原有的信息会出现多次重复的计算，对计算资源是非常大的浪费。由于笔者的遍历要求计算任意起讫点之间的数据和，计算量非常大，笔者需要做一些准备。初次完成关于所有的循环的实现之后，很容易发现 3 重循环嵌套内部求和过程有大量的重复运算。这种独立的计算与循环是独立的，完全可以将求和过程放在 2 重循环之外，使用原有的素数数组进行储存，达到以空间换时间的目的。

这样就省去了重复的计算过程。

2. 遍历机制。

笔者使用的是双重循环结构。双重循环结构能够很好地避免重复的过程，条理非常清晰，但是在多次的遍历中，会有很多的数据被遍历过很多次。避免了数据的重复但是加重了操作的重复性，浪费了计算资源，好在本次实验中 2 重循环是完全合理的，不会超时。笔者这里想到的优化方式是使用 KMP 算法，标记检查位，消除重复次数，节约计算资源。至于二分法也是不错的选择，但是实现需要使用递归，最后 debug 的时候也有诸多问题，递归难于 debug。

3. 配图：



smie15352015

where there is will, there is a way.

[Logout](#)[Home](#) [Problems](#) [Contests](#) [Courses](#) [Ranklist](#) [Submit](#) [Setting](#) [Status](#) [Discuss](#)

<->

[Problem](#)
[Description](#)**Solved****Number**

2246

Submit**Number**

4218

 Problem
solved[Statistics](#)[Source code](#)[Discuss](#)[Submit](#)

1259. Sum of Consecutive Primes

Constraints

Time Limit: 1 secs, Memory Limit: 32 MB

Description

Some positive integers can be represented by a sum of one or more consecutive prime numbers. How many such representations does a given positive integer have? For example, the integer 53 has two representations $5 + 7 + 11 + 13 + 17$ and 53. The

附录、提交文件清单

15352015-caogj-1259-v0

1259.cpp;