

C趣味程序百例(09)要发就发 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/135/2021_2022_C_E8_B6_A3_E5_91_B3_E7_A8_8B_c97_135314.htm 32.要发就发 “ 1898--要发就发 ”。

请将不超过1993的所有素数从小到大排成第一行，第二行上的每个素数都等于它右肩上的素数之差。编程求出：第二行数中是否存在这样的若干个连续的整数，它们的和恰好是1898？假好存在的话，又有几种这样的情况？
第一行：2 3 5 7 11 13 17.....1979 1987 1993
第二行：1 2 2 4 2 4..... 8 6

*问题分析与算法设计：首先从数学上分析该问题：假设第一行中的素数为 $n[1]$ 、 $n[2]$ 、 $n[3]$... $n[i]$ 、...第二行中的差值为 $m[1]$ 、 $m[2]$ 、 $m[3]$... $m[j]$...。其中 $m[j]$ 为： $m[j]=n[j+1]-n[j]$ 。则第二行连续N个数的和为： $SUM=m[1] m[2] m[3] \dots m[j]$

$=(n[2]-n[1]) (n[3]-n[2]) (n[4]-n[3]) \dots (n[j+1]-n[j]) =n[j+1]-n[1]$
由此题目就变成了：在不超过1993的所有素数中是否存在这样两个素数，它们的差恰好是1898。若存在，则第二行中必有所需整数序列，其和恰为1898，。对等价问题的求解是比较简单的。由分析可知，在素数序列中不必包含2，因为任意素数与2的差一定为奇数，所以不必考虑。 *程序与程序注释：

```
#include #include #define NUM 320 int number[NUM]. /*存放不超过1993的全部奇数*/ int fflag(int i). void main() { int i,j,count=0. printf("there are follwing primes sequences in first row:\n"). for(j=0,i=3.i if(fflag(i)) number[j ]=i. for(j--.number[j]>1898.j--) /*从最大的素数开始向1898搜索*/ { for(i=0.number[j]-number[i]>1898.i ). /*循环查找满足条件的素数*/ if(number[j]-number[i]==1898) /*若两个素数的差为1898，
```

```
则输出*/ printf("(%d).=,.....,%d\n", count,number[i],number[j]). }  
} 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com
```