

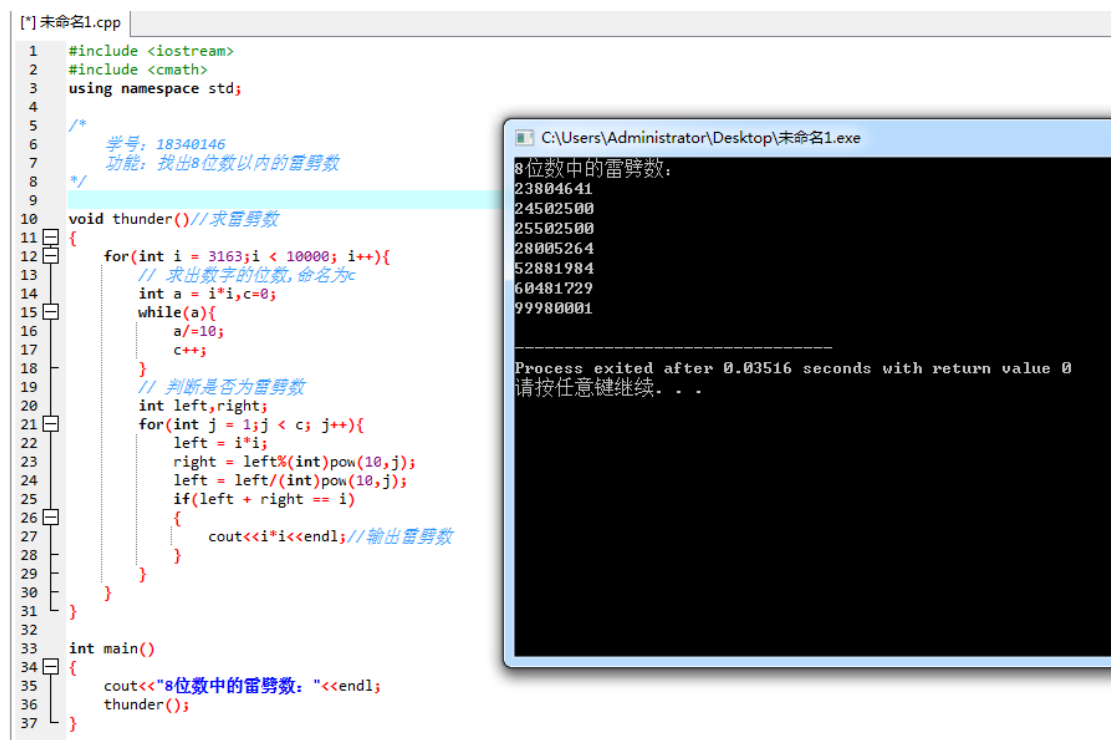
18340146_宋渝杰_实验 1

实验一:雷劈数

实验思路:

根据雷劈数的原理,通过遍历 10000000 到 99999999 内的所有完全平方数的方式,将该数字按所有可能性分成左右两个新数字,判断相加再平方之后是否为两数连接,是的话就为雷劈数,并将其输出.

实验代码及结果截图:



```
[*] 未命名1.cpp
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  /*
6   学号: 18340146
7   功能: 找出8位数以内的雷劈数
8  */
9
10 void thunder()//求雷劈数
11 {
12     for(int i = 3163; i < 10000; i++){
13         // 求出数字的位数,命名为c
14         int a = i*i, c=0;
15         while(a){
16             a/=10;
17             c++;
18         }
19         // 判断是否为雷劈数
20         int left, right;
21         for(int j = 1; j < c; j++){
22             left = i*i;
23             right = left%(int)pow(10, j);
24             left = left/(int)pow(10, j);
25             if(left + right == i)
26             {
27                 cout<<i*i<<endl; //输出雷劈数
28             }
29         }
30     }
31 }
32
33 int main()
34 {
35     cout<<"8位数中的雷劈数: "<<endl;
36     thunder();
37 }
```

```
C:\Users\Administrator\Desktop\未命名1.exe
8位数中的雷劈数:
23804641
24502500
25502500
28005264
52881984
60481729
99980001

-----
Process exited after 0.03516 seconds with return value 0
请按任意键继续. . .
```

实验二:幻阵

(a)判断幻阵

实验思路:

输入矩阵之后,通过遍历每行每列和对角线的和是否为同一个数,

是的话就为幻阵,输出 “Yes” ,否则输出 “No”

实验结果截图:

```
请输入矩阵的大小:
3
请输入矩阵:
8 1 6
3 5 7
4 9 2
Yes!

-----
Process exited after 8.485 seconds with return value 0
请按任意键继续. . .
```

```
请输入矩阵的大小:
3
请输入矩阵:
2 3 4
5 6 7
8 9 1
No!

-----
Process exited after 8.642 seconds with return value 0
请按任意键继续. . .
```

(b)输出幻阵

实验思路:

输入矩阵的大小, 根据奇数幻阵的排列特性进行输出即可。(时间问题, 暂不能实现偶数幻阵, 无输出)

算法:

从第一行中心位置为开始, 右上角为数字+1, 如右上角已有数据则改为正下方, 直到填满整个矩阵。

实验结果截图:

```
输入矩阵大小:
3
8 1 6
3 5 7
4 9 2

-----
Process exited after 2.146 seconds with return value 0
请按任意键继续. . .
```

```
输入矩阵大小:
7
30 39 48 1 10 19 28
38 47 7 9 18 27 29
46 6 8 17 26 35 37
5 14 16 25 34 36 45
13 15 24 33 42 44 4
21 23 32 41 43 3 12
22 31 40 49 2 11 20

-----
Process exited after 2.204 seconds with return value 0
请按任意键继续. . .
```