

首先可以根据错误类型作出大致的判断:

譬如 Presentation Error, 这个说明基本是正确答案了, 不过中间输出的时候多了空格回车这类的

Runtime Error, 这个一般是数组越界导致的好像

Output Limit exceed 这个可能是输入多组数据的判断有问题, 或者哪里输出了多余的东西导致输出过多

poj 网站的 FAQ 都有说明的

普遍情况是 Wrong Answer

想各种数据测试

譬如 3180m-n 这道题目, 我能想到的只有

$m < n$

$m = n$

$m - n$ 的结果前面有冗余的 0 需要去掉

数据大部分是随机生成的, 有些是手动调整, 为了达到上面这三条的测试。

随便试验能找到 bug 比较少, 还是努力让自己想全面一点比较好。

20071026

一、 给定一个二维 (5*5) 数组, 求鞍点所在的位置。鞍点是一个元素, 它是所在行的最大值, 并且是所在列的最小值 (第 4 行第 1 列的元素就是鞍点, 值为 8)。找到鞍点, 则输出行和列以及鞍点的值。不存在鞍点时, 要输出 “not found”。此题提交给助教。(假设每行只有一个最大的值, 每列只有一个最小值)。。

```
int a[5][5]={ 11, 3, 5, 6, 9,
              12, 4, 7, 8, 10,
              10, 5, 6, 9, 11,
              8, 6, 4, 7, 2,
              15, 10, 11, 20, 25};
```

```
void f1()
{ int a[5][5]={11, 3, 5, 7, 9, 12, 4, 6, 8, 10, 10, 3, 6, 9, 11, 8, 6, 4, 7, 2, 15, 10, 15, 20, 25};
  int i, j, k, max, column, flag = 0;
  for (i=0; i<5; i++)
  { for (j=0; j<5; j++)
    { cout<<setw(4)<<a[i][j];
      cout<<"\n";
    }
    for (i=0; i<5; i++)
    { max=a[i][0]; column=0;
```

```

for (j=0;j<5;j++)
    if (a[i][j]>max)
    {
        max=a[i][j];column=j;
    }
for (k=0;k<5;k++)
    if (a[k][column]<a[i][column]) break; //a[i][column]是当前行最大的
if (j==5)
{
    cout<<"第 "<<i+1<<" 行、,第" <<column+1<<" 列元素 "
        <<a[i][column]<<"是鞍点"<<endl;
    flag = 1
}
}
if (!flag) cout << "not found" << endl;
}

```

二、校田径运动会上, A, B, C, D, E 分获 100 米(1), 400 米(2), 跳高(3), 跳远(4)和三级跳冠军(5)

甲: B 获得三级跳冠军, D 获得跳高冠军

乙: A 获的百米冠军, E 获得跳高冠军

丙: C 获得跳远冠军, D 获得四百米冠军

丁: B 获得跳高冠军, E 获得三级冠军

情况是每个人说对一句, 说错一句, 问 A, B, C, D, E 分别获得了哪项冠军?

```

void f2 ()
{
    int A, B, C, D, E, Jia, Yi, Bing, Ding;
    char program[6][25]={" ", "100m", "400m", "jump high", "jump long", "three step jump long"};
    for(A = 1; A <= 5; A++)
        for (B = 1; B <= 5; B++)
            if( A != B)
                for(C = 1; C <= 5; C++)
                    if((C != A) && (C != B))
                        for (D =1; D <= 5; D++)
                            if((D != A) && (D != B) && (D != C))
                                for (E =1; E <= 5; E++)
                                    if((E != A) && (E != B) && (E != C) && (E != D))
                                        {
                                            Jia = (B == 5) + (D == 3);
                                            Yi = (A == 1) + (E == 3);
                                            Bing = (C == 4) + (D == 2);
                                        }
}

```

```

Ding = (B == 3) + (E == 5);
if (Jia==1 && Yi ==1 && Bing == 1 && Ding == 1)
{
    cout << "A = " << program[A] << endl;
    cout << "B = " << program[B] << endl;
    cout << "C = " << program[C] << endl;
    cout << "D = " << program[D] << endl;
    cout << "E = " << program[E] << endl;
}
}

```

}

三、输入一个英文句子，长度不超过 40 个字符。编写程序，输出句子中最长的一个单词。此题提交 ACM, 题号 3179, 不需要提交给助教

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    const int strSize = 200;
```

```
    int i, word = 0, max = 0, count = 0, p, len;
```

```
    char str[strSize + 1];
```

```
    while(cin.getline(str, strSize + 1))
```

```
    {
```

```
        len = strlen(str);
```

```
        word = 0; max = 0; count = 0;
```

```
        for (i = 0; i <= len; i++)
```

```
        {
```

```
            if ((str[i]==' ') || (str[i]==',' ) || (str[i]=='.' ) || (str[i] =='\0'))
```

```
                if(word == 1) //分隔符前面是字符，则结束一个单词
```

```
                {
```

```
                    if (count >= max) //当前单词长度更长
```

```
                    {
```

```
                        max = count; p = i - count; //p 记录最长单词的起始位置
```

```
                    }
```

```
                }
```

```
                word=0;count=0; //一个单词处理完毕。
```

```
            }
```

```
            else
```

```
            {
```

```
                count++; word = 1; //读到一个字符，长度加 1
```

```
            }
```

```
        }
```

```
{
```

```

        for (int j = 0; j < max; j++)           //从起始位输出最长的字符
            cout << str[p + j];
        cout << endl;
    }
}

```

四、编写一个程序，输入是两个无符号整数 m 和 n（假定 $m > n$ ）。计算并输出 m 和 n 的和（其中 m 和 n 最多可以是 200 位十进制数，以字符形式输入）

此题提交 ACM, 题号 3180, 不需要提交给助教

```

#include <iostream>
#include <cstring>

using namespace std;

int main()
{
    const int MAX_LEN = 200;
    int an1[MAX_LEN]; int an2[MAX_LEN];
    char seLine1[MAX_LEN + 1]; char seLine2[MAX_LEN + 1];
    while (cin.getline(seLine1, MAX_LEN + 1))
    {
        cin.getline(seLine2, MAX_LEN + 1);
        int nLen1 = strlen(seLine1);
        int nLen2 = strlen(seLine2);

        memset(an1, 0, sizeof(an1));
        memset(an2, 0, sizeof(an2));

        int i, j=0;
        for (i = nLen1-1; i >= 0; i--)
            an1[j++] = seLine1[i] - '0'; //将字符数组 1 变成整数数组，并倒置

        j=0;
        for (i = nLen2-1; i >= 0; i--)
            an2[j++] = seLine2[i] - '0'; //将字符数组 2 变成整数数组，并倒置

        int sign = 0; //初值假设结果为正数

        i = strcmp(seLine1, seLine2);
        if (i == 0) //两个数相等，输出 0
        {
            cout << 0 << endl;
            continue;
        }
    }
}

```

```

}

i = nLen1 >= nLen2 ? (nLen1 - 1) : (nLen2 - 1); //定位较长的数组的最高位
while(an1[i] == an2[i]) i--; //从最高位，找到第一个不相等的两个数
if (an1[i] < an2[i]) //被减数小于减数，互换元素
{
    sign = 1; int t;
    for (i = 0; i <= nLen2 - 1; i++)
    {
        t = an1[i];
        an1[i] = an2[i];
        an2[i] = t;
    }
}

for (i = 0; i < MAX_LEN ;i++) //直接按位相减
{
    if(an1[i] >= an2[i])
    {
        an1[i] -= an2[i];
    }
    else
    {
        an1[i] += 10;
        an1[i + 1]--; //这个地方可能会小于0，因此数组要整数。
        an1[i] -= an2[i];
    }
}

if (sign == 1) //是负数，先输出负号
    cout << "-";

i = MAX_LEN - 1;
while(an1[i]==0) i--; //找到第一个不为0的位
for(;i >= 0; i--)
    cout << an1[i]; //输出每一位数
cout << endl; //ACM 需要
}
}

```

3179 的部分测试数据

For detailed information on the university graduate programs, please use the list below to view the website of the specific academic college or department that interests you.

a b c d..... ..

You, a good man, is not bad .

Hello.

Robert E. Chambers Jr., who killed a young woman in Central Park in 1986, is now accused of dealing cocaine with his companion from a Midtown apartment. He is really a bad guy in my opinion.....

You,

3180 的部分测试数据

- 5555
- 5554

- 9999
- 4567

- 0
- 1

- 47366160654413407880568388379428403170145870123781677450124969190655654154586931753445801796053958984377544538474996479982005
- 11840536408284598554169714235348851697216919690108976971059719870357177064815863658792410689262855007492377365883860903624569

- 652502026411765071877762775977302767628374781493295642684305954698056563070896968963965086765937075784105083735306
- 337138144832768792165276275260333248480697188730005257403892948689044734018601901309228130804064654673055513547853

- 9032
- 6761

- 271480392629626291126732171008789945622332652561651102209696073258512764422846021267927556534062411213306082219288908336940222772135122043572194
- 320639194772681272363363611065329482018990708850040773017252377496870407221944039948701190129789951236056145044954605009720434832582535595185168

- 64873627134847026974648922537027316155995786578773593658385163539667
- 66070329210194160498420073536422452147479586882587515135794988329347

- 44410342245761374356860423710292120434496287830968680613905835803070624541853431843865658903946900423177059249872440424425038130065615263972229270489597887078319
- 48490587233466228565130584608423575720514271477262996692662288794217955900626606145002942701174751021672369071111563762628247534683088057617614098092026

677845673

- 20698106625608233815053235231151
- 81439712240078732780970700808528

- 3135546466224048013490158330259467345503083395674687402326090797931231049826
8149836557244995985421565956486899835208921937721809882216301170885048822095
357532032205288529367255584865510328163813
- 6040594735244264102596686743855099379545199051059506136293977507679662847175
4487949249578787660692922379255102523408686571818533075933005023914636721372
866788347076533925875939039589646657523040

- 5589933141268625743248657134836820892096893254721382666684430259100759146822
38501754387560843609098568878204583532910025458270
- 8016842546913151822626167058570020035153823848399207772331178204661485734527
81507665219823068877117037127200233190080569769533

- 905612936063877847298918281530854290521647
- 587693942738214490067336015104072198045840

- 54892995910425067104304162837126170102197996513963529302088
- 57348872315640218546

- 3434129697047940169709363421661694178411529371193153678979414799659879146907
485452437668789185013429103213097055550389079485044556782922
- 9973250009706828195956369367665216958964934066255815115842586388258911381577
0617772374379972852456176982316288636870909196171052

- 7345251835834769521356377308777039005088325886308118153560634074040686419469
98760065188050184368
- 7536128913995141858934903342541230999029760087283087954123460932148197617872
5818750720223742009171922139089763931703488758783405940839964467

- 157273929470720466559259724325091290883911649955604
- 7207272250170221574435885218358207187793928135868325999444466743959568561406
04633054689970522620149393038048206055957069578265069092959

- 19254049858470488266
- 9018710864354097349506750431161976761773841557461215354810274095829069254314
1298886281730815410766510689767815783030860256080749949816942782974979522160
76257180689046352440680519367260926575738700606

- 9796281933855353744343141593451595450389985842704801012324640781926217797247
07849760770920966716295316063392387003983184532755938532125192

