

# 2014 级 C++第六次上机

## A. jhljx 学数组

### Problem Description

终于到数组啦。。呵呵呵。。

jhljx 决定开始学习数组了。。

一天老师给了他一个字符串。。字符串吖。。能吃吗？

老师让你统计字符串中每个字符的个数。是不是很简单吖。

### Input

输入多组数据。

每组数据为一个字符串，保证字符串的长度小于 1000。字符串中保证只有英文字母。

### Output

输出字符串中出现的字符的个数。请按照先输出小写字母的个数，再输出大写字母的个数的顺序输出。每组数据间用一个空行隔开。详细请见样例。

### Sample Input

```
AbccbA
abcdeBCCDE
```

### Sample Output

```
# # #
# # #
[b] [c] [A]

#
# # # # # # # #
[a] [b] [c] [d] [e] [B] [C] [D] [E]
```

### Hint

C 语言使用字符串处理的库函数需要添加 `string.h` 头文件

即`#include<string.h>`

C 语言请用以下形式输入

```

char a[1010];
while(gets(a))
{
    int k=strlen(a); //计算字符串长度的函数, k 表示字符串的长度
    .....
}

C++语言使用字符串处理的库函数需要添加 string.h 头文件或者 cstring 头文件
即#include<cstring>或者#include<string.h>都可以
C++语言请用以下形式输入
char a[1010];
while(cin>>a)
{
    int k=strlen(a); //计算字符串长度的函数, k 表示字符串的长度
    .....
}

注意每组数据的输出结果之间有一个空行。
未出现的字符不要输出。

```

这个题统计不难。难在输出。我在注释中说明我是怎么输出的。

```

#include <iostream>

#include<cstdio>
#include<cstring>
using namespace std;
const int maxn=1000100;
char a[maxn];
int main()
{
    while (gets(a)) /*学长教的。但是参考书上写这个函数已经被废除了。不
    推荐使用*/
    {
        int max=-2; /*max 用来保存最大值。因为统计出现次数最小是 0. 所以
        取个负数。*/
        int tj[100]={0}; //这个是用来统计 52 个字母出现的次数的
        int k=strlen(a);
        for (int i=0;i<k;i++) //从第一个字符到最后一个
        {
            int nv=a[i]; //类型转换一下。
            //小写字母从 97 开始。大写从 65 开始。

```

```

        //cout <<nv<<endl; //这是输出临时变量时用的。（检验）
        if (nv>=97&&nv<=122)
            tj[nv-58]++;
//因为他要求小写字母在前，另外为了保证连续。就减掉 58、这样就是从 39 到
//64 是小写的 a 到 z。65 到 90 是大写的 A 到 Z。
        else if(nv>=65&&nv<=90)
            tj[nv]++;
    }
    for (int i=39;i<=90;i++) //找到最大值。
    {
        //cout <<tj[i]<<endl;
        if (max<tj[i])
            max=tj[i];
    }
    // cout <<max<<endl; //检验。请无视。
    // getchar();
    for (int j=0;j<max;j++)
//这是从 39 (a) 开始。依次去查看是否大于等于最大值。是否值为 0
//如果值为 0，就不输出。大于等于最大值就输出一个#。小于就输出相应的空
//格。。。
    {
        for (int i=39;i<=90;i++)
        {
            if (tj[i]!=0&&tj[i]>=max-j)
//让 max 递减 1。。其实可以在第一个 for 循环里进行的。。
//for(j=max;j>0;j--)    if (tj[i]!=0&&tj[i]>= j)
            {
                cout <<" # ";
            }
            else if(tj[i]!=0)
                cout <<" ";
            }
            cout <<endl;
        }
    }
//到这里#的输出已经结束了。下面输出[a]这个东东
    for(int i=39;i<=90;i++)
    {
        if (tj[i]!=0) //没出现的不输出。。。
        {
            char g;
//再把 g 变回来。。就是把减掉的加回去。
            if(i<65)
                g=i+58;
            else

```

```

        g=i;
        cout <<"["<<g<<"]";
    }
}
cout <<endl<<endl;
}
return 0;
}

```

## B. 巫女的怀胎

### Problem Description

---

若非技术，那便是才能。

-----  
 比如  $l_jx$  在算术方面具有卓越的才能。

已知  $x$  加上  $x$  的各个数字之和为  $y$ ，则说明  $x$  和  $y$  是一对  $cp$ 。给出  $n(1 \leq n \leq 100000)$ 。  
 求  $n$  的最小  $cp$ 。无解输出 0。

### Input

---

输入多组数据。（小于 100 组）  
 每组数据一行，为一个数  $n$ 。( $1 \leq n \leq 100000$ )

### Output

---

每组数据输出一行。为  $n$  的  $cp$ 。

### Sample Input

---

```

216
121

```

# Sample Ouput

---

198  
0

在这解释一下。

一个数的 **cp** 是有最小可能值的。因为各个位加起来有一个最大数。举例如下：

3456

比他小的数的各个位加起来最大是： $2+9+9+9=29$ ；所以。他的 **cp** 最小可能数不会小于  $3456-29=3427$ ；

所以用 **for** 循环来做的话。循环次数是很少的。而不是从 1 开始。。。

```
#include <iostream>
#include<cstdio>
#include<cstring>
#include<cmath>
using namespace std;
int main()
{
    int n,ws,sum,man,a[10],pd,rus;
    while (cin>>n)
    {
        pd=0;
        ws=log10(n);//n 的位数-1
        // cout <<ws<<endl;
        man=ws*9+n/pow(10.0,ws*1.0)-1;
        //计算各个位相加的最大值。就是（位数-1）乘以 9+最高位数减一
        //cout <<man<<endl;
        for (int i=n-man-1;i<n;i++)
        //开始从可能的最小值开始循环。。剩下的略。
        {
            sum=i;
            int tem=i;
            while (tem>0)
            {
                sum+=tem%10;
```

```
        tem/=10;
    }
    rus=i;
    if (sum==n)
    {cout <<rus<<endl;
    pd=1;
    break;
    }
    else continue ;
}
if (pd==0)
cout <<0<<endl;
}
return 0;
}
```

## C. 善恶的彼岸

### Problem Description

---

You guys do not notice that we are gifted just for being humans. We are absolute predators, we do not even have any enemies. Maybe there are animals watching us and thinking that "Someday we will BEAT THEM DOWN!". 但是 "Someday we will BEAT THEM DOWN!" 这种格式会导致让人分不出是左引号还是右引号，所以我们需要把它变为“”。一句话：把"变为“”。

### Input

---

输入一篇文章。

### Output

---

输出引号转换后的文章。

## Sample Input

---

"To be or not to be," quoth the Bard, "that is the question".

## Sample Output

---

“To be or not to be,” quoth the Bard, “that is the question”.

## Hint

---

注意是将英文引号转为中文引号。

这个题。没啥好说的吧。。注意一下上下引号的区分就好了。。

```
#include <iostream>
#include<cstdio>
#include<cstring>
using namespace std;
const int maxn=1000100;
char a[maxn];
int up[maxn]={0};
int main()
{
    int upd=1;
    while (gets(a))
    {
        upd=1;
        int k=strlen(a);
        for (int i=0;i<k;i++)
        {
            if(a[i]=='"' &&upd==1)
            {
                up[i]=1;
            }
        }
    }
}
```

```

        upd=2;
    }
    else if(a[i]==' ' && upd==2)
    {
        up[i]=2;
        upd=1;
    }
}
for (int i=0;i<k;i++)
{
    if(up[i]==1)
        cout << " ";
    else if (up[i]==2)
        cout << " " " ";
    else
        cout << a[i];
}
cout << endl;
}
return 0;
}

```

## D. jhljx 分解质因数

### Problem Description

---

jhljx 最近在学小学数学，老师教他分解质因数。也就是说给你一个数  $n$ , 让你把它分解成若干个质数的乘积的形式。

### Input

---

输入多组数据。

每组数据一行，为一个数  $n$ 。( $1 \leq n \leq 10^6$ )



# Output

---

每组数据输出一行。为一个表达式。表达式中的数字必须按从小到大的顺序排列。（详见样例）

## Sample Input

---

11  
9412

## Sample Output

---

11  
2\*2\*13\*181

## Hint

---

注意可能木有乘号哦。。  
请用 `scanf` 和 `printf` 输入输出。

```
#include <iostream>

#include<cstdio>

#include<cstring>

#include<cmath>

using namespace std;

int main()

{

    int n, ret, js, enf, pd1, enh, n1, i;
```

```
while (cin>>n)
```

```
{
```

```
    js=0;
```

```
    if (n==1)
```

```
        cout <<1<<endl;
```

//1 单独讨论一下。

```
    else
```

```
    {
```

```
        for ( i=2;n>1;i++)
```

//从 2 开始。

```
    {
```

```
        while (n%i==0)
```

/\*如果 n 是 i 的倍数。就输出一个 i 并让 n=n/i;就是把 n 除以了 i 直到除不动。  
所以不会出现 i 的倍数满足被 n 整除了。所以一定是质数。。直到 n 不是 i 的倍  
数。。就把 i 加一。再循环。\*/

```
    {
```

```
        if (js!=0)
```

```
            cout <<"*";
```

//js 作为输出的计数。。如果不是第一次输出，就输出一个\*

```
            cout <<i;
```

```
            js++;
```

```
            n/=i;
```

```
    }
```

```

    }

    cout <<endl;

}

}

return 0;

}

```

## E. jhljx 学排列组合

### Problem Description

---

jhljx 最近又学习了一些排列组合的知识。比如什么隔板法，插空法神马的。。相信你们都会吧。。毕竟 **College Entrance Examination** 的数学考试中考过。。

于是 jhljx 决定来考考你。。你要是记不住的话去找高中数学老师吧。。~\((\geq \nabla \leq)/\sim 啦啦啦你看我把题目的背景都告诉你了。。相信你能 AC 哟。。

快到圣诞节了。。于是松辰学妹送给他了一些糖果。。233

松辰学妹总共送了  $n$  种糖果。每种糖果的个数为  $a[1], a[2], a[3], \dots, a[n]$ 。jhljx 讨厌连续吃两块相同种类的糖。。所以他会尽量避免吃到同种类的糖。

为啥捏？因为他是强迫症吖。。上一次上机他的强迫症不就犯了？请猛戳这里->[jhljx 的强迫症](#)

所以请你判断他是否会吃到同种类的糖。

### Input

---

输入多组数据。

每组数据两行，第一行为一个数  $n(1 \leq n \leq 10000)$ ，第二行为  $n$  个数，每个数  $a[i]$  表示  $i$  种类的糖共有多少个。(保证  $a[i]$  在 `int` 范围内)

### Output

---

每组数据输出一行。如果可以避免吃到同种类的糖，输出 **Yes**, 如果无法避免，输出 **No**。

## Sample Input

---

3  
4 1 1

## Sample Output

---

No

额。这个就是算一下最大值的 2 倍和总数的大小关系。。。

坑人的是爆掉了 int。和。输出 **YeS**,如果无法避免，输出 **No**。

```
#include<stdio>
#include<cstring>
int main()
{
    int a,max,js;
    long long sum;
    int n;
    while(scanf("%d",&n)!=EOF)
    {
        js=0;
        sum=0;

        for (int i=0;i<n;i++)
        {
            scanf("%d",&a);
            if(js==0)
                max=a;
            else
            {
                if (max<a)
                    max=a;
            }
            sum+=a;
            js++;
        }
        if(sum+1>=max*2)
```

```
        printf("YeS\n");
    else
        printf("No\n");
}
return 0;
}
```

## F. jhljx 学画画

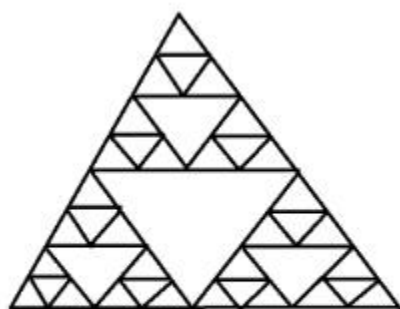
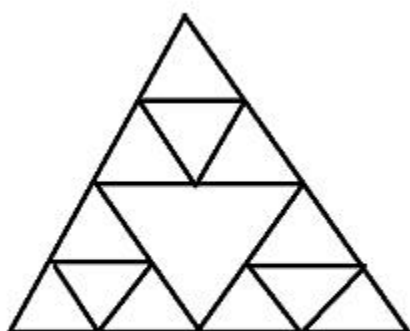
### Problem Description

---

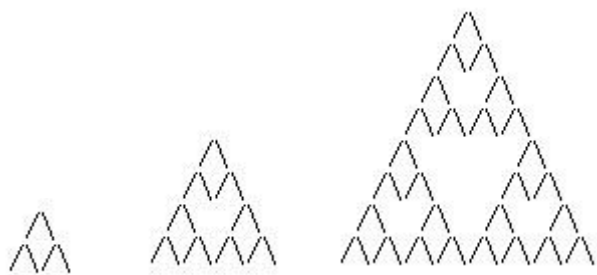
jhljx 是一个爱好广泛的人。最近他迷恋上了画画。还记得上次他教松辰学妹学画画。这次他决定自己动手画画啦。。

真是好棒耶。。~\( $\cong \nabla \cong$ )/~啦啦啦

他要画的图形是这样的。。



但是为了简化，我们将图形变成了这样。



# Input

输入多组数据。  
每组数据输入一个 n，表示图形的行数。(保证 n 为 2,4,8,16……，且 n<=10000)

# Output

输出正确的图形。

# Sample Input

8

# Sample Output



啊啊啊啊啊。这个题我竟然做出来了。！！！！

太高兴了。。我是用递归做的。。

```
#include <iostream>

#include<cstdio>

#include<cstring>

//#include<fstream>

#include<cmath>

//#define cout fout

//#define cin fin

using namespace std;

//ofstream fout("e:\\output.txt");

//ifstream fin("e:\\input.txt");

void disp(int n,int hs)//递归的函数。表示图形 n 的第几行。不包含换行符
{
    if (n==2)

//n=2 的基本情况

    {
        if (hs==1)

            cout <<" /\\" <<" ";

//尤其要注意前后都要有空格。要保持队形相同。这样再次调用时不会出偏差

        else if (hs==2)
```

```

        cout << "/\\\\";

    }

    else

//当 n 大于 2 时。规律是前 n/2 行和上一个图形一样（就是和 n/2 的图形），但
//是空格不同。后 n/2 行是上次图形的两次调用。

    {

        if (hs<=n/2)

        {

            for (int i=1;i<=n/2;i++)

                cout <<" ";

//空格数目。仔细算算就知道了。。。

            disp(n/2, hs);

            for (int i=1;i<=n/2;i++)

                cout <<" ";

//务必注意，保持队形的重要性。前后都要有相应的空格

        }

        else

        {

            hs=hs-n/2;

//行数减去多的 n/2 行。

            disp(n/2, hs);

            disp(n/2, hs);

//两次调用

```



```
        }

    }

}

int main()

{

    int n;

    while (cin>>n)

    {

        for (int i=1;i<=n;i++)

            //用 for 循环输出每一行。注意换行。

            {

                disp(n, i);

                cout <<endl;

            }

    }

    return 0;

}
```