

问题 A：求两个数组中相同的元素

时间限制：1 Sec 内存限制：128 MB

提交：282 解决：188

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

输入两个数组（数组元素个数 6 和 8），输出在两个数组中都出现的元素（如 a[6]={2,3,4,5,6,7},b[8]={3,5,7,9,11,13,15,19},则输出 3、5、7）。

输入

输出

样例输入

```
2 3 4 5 6 7
3 5 7 9 11 13 15 19
```

样例输出

```
3
5
7
```

```
#include<stdio.h>
int main()
{
    int a[6],b[8];
    int i,j;
    for(i=0;i<6;i++)
        scanf("%d",&a[i]);
    for(i=0;i<8;i++)
        scanf("%d",&b[i]);
    for(i=0;i<6;i++)
    {
        for(j=0;j<8;j++)
        {
```

```

        if(a[i]==b[j])
            printf("%d\n",a[i]);
    }
}
return 0;
}

```

问题 B: 求数列中大于数列平均值的数的个数

时间限制: 1 Sec 内存限制: 128 MB

提交: 771 解决: 553

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

输入 10 个数，求它们的平均值，并输出大于平均值的数据的个数。

输入

10 个数

输出

大于平均数的个数

样例输入

1 2 3 4 5 6 7 8 9 10

样例输出

5

```

#include<stdio.h>
int main()
{
    int aver,s[10],sum=0,q=0,i;
    for(i=0;i<=9;i++)

```

```
{
    scanf("%d",&s[i]);
    sum=sum+s[i];
}
aver=sum/10;
for(i=0;i<=9;i++)
{
    if(s[i]>aver)++q;
}
printf("%d",q);
return 0;
}
```

问题 C：标识符（循环）

时间限制：1 Sec 内存限制：128 MB

提交：748 解决：225

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

C 语言中的标识符只能由字母、数字和下划线三种字符组成，且第一个字符必须是字母或下划线。c89 规定标识符在 31 个字符以内，且不能是关键字。假设忽略关键字比较，采用 c89 标准。输入一行，判断是否合法标识符。

输入

第一行，测试次数

每组测试数据一行字符

输出

对每组测试数据，若为 C 语言合法标识符，输出 YES，否则输出 NO。

样例输入

```
5
temp
ABC10
area90
9long
width!
```

样例输出

```
YES
NO
YES
NO
NO
```

```
#include<stdio.h>
#include<string.h>
int main()
{
    int t,T,mark,i,len;
    char s[1000];
    scanf("%d",&T);
    getchar();
    for(t=1;t<=T;t++)
    {
        gets(s);
        len=strlen(s);
        mark=0;
        if((s[0]>='a' && s[0]<='z') || (s[0]>='A' && s[0]<='Z') || (s[0]=='_') &
&len<=31)
        {
            for(i=1;i<len;i++)
            {
```

```

        if((s[i]>='a' && s[i]<='z') || (s[i]>='A' && s[i]<='Z'
') || (s[i]=='_')) || (s[i]>='0' && s[i]<='9'))
        {
            mark++;
        }
    }
    if(mark==len-1) {
        printf("YES\n");
    }
    else {
        printf("NO\n");
    }
}
else {
    printf("NO\n");
}
return 0;
}

```

问题 D：提取数字串（数组）

时间限制：1 Sec 内存限制：128 MB

提交：184 解决：41

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

输入多行字符，请将其中的数字串提取出来，并按降序排序。

假设小数点只出现在数字串中；各数字串长度不超过 19；数字串个数不超过 30 个。没有负数。

查：atof 函数。

输入

多行字符

输出

将提取的数字串，按数字的降序排序输出，保留两位小数。具体输出格式见样例。所有提取的数据不超浮点数范围。

样例输入

```
abdsfsd12.34#@fsd654vxvx123.890afsfsd12.fsd fs979867
nmjhf32432gg1.23
fsdfsd42AVSVfd89.90fdfsdfs1
.2fsdfd54
```

样例输出

```
979867.00 32432.00 654.00 123.89 89.90 54.00 42.00
12.34 12.00 1.23 1.20
```

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main()
{
    int m=0,i,n,len,j;
    double b[3000],v;
    char s[1000]={'\0'},s1[1000],a[300][100]={'\0'};
    while((scanf("%s",s1))!=EOF)
    {
        strcat(s,s1);
    }
    len=strlen(s);
    for(i=0;i<len;i++)
    {
        if(s[i]>='0' && s[i]<='9' || (s[i]=='.'))
        {
            n=0;
```

```

        while(s[i]>='0' && s[i]<='9' || (s[i]=='.'))
        {
            a[m][n++]=s[i++];
        }
        m++;
    }
}
for(i=0;i<m;i++)
{
    b[i]=atof(a[i]);
}
for(i=1;i<m;i++)
{
    for(j=0;j<m-i;j++)
    {
        if(b[j]<b[j+1])
        {
            v=b[j];
            b[j]=b[j+1];
            b[j+1]=v;
        }
    }
}
for(i=0;i<m-1;i++)
{
    printf("%.2lf ",b[i]);
}
printf("%.2lf",b[m-1]);
return 0;
}

```

问题 E：矩阵取数问题

时间限制：1 Sec 内存限制：128 MB

提交：267 解决：107

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

一个 $N \times N$ 矩阵中有不同的正整数，经过这个格子，就能获得相应价值的奖励，从左上走到右下，只能向下向右走，求能够获得的最大价值。

例如： 3×3 的方格。

1 3 3

2 1 3

2 2 1

能够获得的最大价值为：11。

提示：定义 $f(x,y)$ 为从左上角 $(0, 0)$ 走到第 x 行第 y 列获得的最大价值。

找出 $f(x,y)$ 与 $f(x-1,y)$ 和 $f(x,y-1)$ 之间的关系。

输入

第 1 行： N ， N 为矩阵的大小。 $(2 \leq N \leq 500)$

第 2 - $N + 1$ 行： 每行 N 个数，中间用空格隔开，对应格子中奖励的价值。 $(1 \leq N[i] \leq 10000)$

输出

输出能够获得的最大价值。

样例输入

```
3
1 3 3
2 1 3
2 2 1
```

样例输出

```
11
```

```
#include <iostream>
#include <cstdlib>
#include <cstdio>
#include <cstring>
#include <algorithm>
using namespace std;

int main() {
    int i, j, n, s[501][501]={0};
    scanf("%d",&n);
    for(i=1;i<=n;i++) {
        for(j=1;j<=n;j++) {
            scanf("%d",&s[i][j]);
            s[i][j]+=max(s[i-1][j],s[i][j-1]);
        }
    }
    printf("%d\n",s[n][n]);
}
```

问题 F：子串统计

时间限制：1 Sec 内存限制：128 MB

提交：159 解决：127

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

用指针实现计算字符串中子串出现的次数

输入

2 个字符串 a 和 b，每个字符串长度不超过 20 个字符

输出

a 中出现 b 的个数

样例输入

```
abcdefbc  
bc
```

样例输出

```
2
```

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main()
{
    char array[100], even[100];
    int i, j, m, n, num=0;
    gets(array);
    gets(even);
    m=strlen(array);
    n=strlen(even);
    for(i=0; i<=m-n; i++)
    {
        if(strncmp(even, &array[i], n)==0)
            num++;
    }
    printf("%d\n", num);
    return 0;
}
```

问题 G: 编写删除字符串中某个字符的函数——函数

时间限制: 1 Sec 内存限制: 128 MB

提交: 1579 解决: 814

[\[提交\]](#) [\[状态\]](#) [\[讨论版\]](#)

题目描述

编制函数 del_char

函数原型为 void del_char(char *,char),函数的功能是删除 a 指向的字符串中值为 ch 的字符, 例如从字符串"AscADef"中删除'A'后, 字符串为"scDef"。

输入

需要删除的字符 ch

需要处理的字符串

输出

处理后的字符串

样例输入

```
A
AscADef
```

样例输出

```
scDef
```

```
#include<stdio.h>
#include<string.h>
void del_char(char *,char);
```

```
int main()
{
```

```

        char c=getchar();
        getchar();
        char str1[1000];
        gets(str1);
        del_char(str1,c);
        puts(str1);
        return 0;
    }
    void del_char(char str[], char c) {
        for(int i=0;i<strlen(str);i++) {
            while (str[i]==c) {
                for (int j=i;j<strlen(str);j++) {
                    str[j]=str[j+1];
                }
            }
        }
    }
}

```