目　　录

http://acm.hrbeu.edu.cn/index.php?act=problemlist

1004、Hay Points

```
#include <stdio.h>
#include <string.h>
int main(){
    char a[1000][17];
    int b[1000];
    char s[100];
    int i,j;
    long sum;
    int m,n;
    scanf("%d%d",&m,&n);
    for(i=0;i<m;i++){
        scanf("%s",a[i]);
        scanf("%d",&b[i]);
    }
    for(i=0;i<n;i++){
        sum=0;
        scanf("%s",s);
        while(s[0]!='.') {
            for(j=0;j<m;j++){
                if(strcmp(a[j],s)==0)sum+=b[j];
            }
            scanf("%s",s);
```

```
        }
        printf("%ld\n",sum);
    }
    return 0;
}
```

1006、 Beavergnaw

```
#include <iostream>
#include <cmath>
#include <iomanip>
using namespace std;
int main() {
    double dd,v,t;
    double d;
    while(1) {
        cin>>dd>>v;
        if(dd==0 && v==0) break;
        t=dd*dd*dd-6.0*v/3.1415926535897932;
        d=pow(t,1.0/3);
        cout<<setprecision(3)<<setiosflags(ios::fixed)<<d<<resetiosflags(ios::fixed)<<endl;
    }
    return 0;
}
```

1007、Power Strings（一）(0.06　5108)　（字符串权）

```
#include <stdio.h>

char s[1000001];

int next[1000001];

int main(){

    int i,j,n,t;

    while(gets(s) && s[0]!='.'){

        i=0;

        next[0]=-1;

        j=-1;

        while(s[i]!='\0'){

            if(j==-1 || s[i]==s[j]){

                ++i;

                ++j;

                next[i]=j;

                continue;

            }

            j=next[j];

        }

        n=i;

        t=n-next[n];

        if(n%t==0)printf("%d\n",n/t);

        else printf("1\n");
```

```
    }
    return 0;
}
```

1017、Above Average

```
#include <iostream>
using namespace std;
int main(){
    double a[1000],sum,num;;
    int t,n,i;
    cin>>t;
    for(;t>0;t--){
        cin>>n;
        sum=0;num=0;
        for(i=0;i<n;i++){
            cin>>a[i];
            sum+=a[i];
        }
        sum/=n;
        for(i=0;i<n;i++)
            if(a[i]>sum)++num;
        printf("%.3lf\%%\n",100.0*num/n);
    }
    return 0;
```

```
}
```

1008、Relatives

```
#include <iostream>
using namespace std;
int eular(int n){
    int ret=1;
    int i;
    for(i=2;i*i<=n;i++){
        if(n%i==0){
            ret*=i-1;
            n=n/i;
            while(n%i==0){
                ret*=i;
                n=n/i;
            }
        }
    }
    if(n>1)ret*=n-1;
    return ret;
}
int main(){
    int n;
    while(cin>>n){
```

```
        if(n==0)break;
        cout<<eular(n)<<endl;
    }
    return 0;
}
```

1010、Tic Tac Toe

```
#include <iostream>
using namespace std;
char s[3][4];
int num(char c){
    int i,j;
    int sum=0;
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            if(s[i][j]==c)++sum;
    return sum;
}
int line(char c){
    int i;
    for(i=0;i<3;i++){
        if(s[i][0]==s[i][1] && s[i][0]==s[i][2] && s[i][0]==c)return 1;
        if(s[0][i]==s[1][i] && s[0][i]==s[2][i] && s[0][i]==c)return 1;
    }
```

```
    if(s[0][0]==s[1][1] && s[0][0]==s[2][2] && s[0][0]==c)return 1;
    if(s[0][2]==s[1][1] && s[2][0]==s[1][1] && s[1][1]==c)return 1;
    return 0;
}

int main(){
    int a,b;
    int n;
    cin>>n;
    for(;n>0;n--){
        getchar();
        gets(s[0]);
        gets(s[1]);
        gets(s[2]);
        a=num('X');b=num('O');
        if(a<b) {
            cout<<"no"<<endl;
            continue;
        }
        if(a-b>1){
            cout<<"no"<<endl;
            continue;
        }
```

```
            if(a-b==1){
                if(line('O')){
                    cout<<"no"<<endl;
                    continue;
                }
            }
            if(a==b){
                if(line('X')){
                    cout<<"no"<<endl;
                    continue;
                }
            }
            cout<<"yes"<<endl;
        }
        return 1;
}
```

1023、 Rock, Scissors, Paper（石头，剪刀，布）

```
#include <iostream>
```

```
using namespace std;

int main() {

	int t,k,p;	int i,j,n;	int r,s;

	char a[102][102],b[102][102];

	for(i=0;i<102;i++){

		a[i][0]='e'; a[0][i]='e';

		b[i][0]='e'; b[0][i]='e';

	}

	cin>>t;

	for(k=1;k<=t;k++) {

		cin>>r>>s>>n;

		for(i=1;i<=r;i++)

			for(j=1;j<=s;j++){

				b[i][j]='e';

				cin>>a[i][j];

			}

		if(n==0){

			for(i=1;i<=r;i++){

				for(j=1;j<=s;j++)

					cout<<a[i][j];

				cout<<endl;

			}

			continue;
```

```
}
for(i=0;i<=r+1;i++){
        a[i][s+1]='e';       a[i][s+1]='e';
}
for(i=0;i<=s+1;i++){
        a[r+1][i]='e';       a[r+1][i]='e';
}
for(i=1;i<=r;i++)
        for(j=1;j<=s;j++){
                if(a[i][j]=='R') {
                        if(a[i-1][j]=='S')b[i-1][j]='R';
                        if(a[i+1][j]=='S')b[i+1][j]='R';
                        if(a[i][j-1]=='S')b[i][j-1]='R';
                        if(a[i][j+1]=='S')b[i][j+1]='R';
                }
                if(a[i][j]=='S') {
                        if(a[i-1][j]=='P')b[i-1][j]='S';
                        if(a[i+1][j]=='P')b[i+1][j]='S';
                        if(a[i][j-1]=='P')b[i][j-1]='S';
                        if(a[i][j+1]=='P')b[i][j+1]='S';
                }
                if(a[i][j]=='P') {
                        if(a[i-1][j]=='R')b[i-1][j]='P';
```

```
                if(a[i+1][j]=='R')b[i+1][j]='P';
                if(a[i][j-1]=='R')b[i][j-1]='P';
                if(a[i][j+1]=='R')b[i][j+1]='P';
            }
        }
for(p=1;p<n;p++) {
        for(i=1;i<=r;i++)
            for(j=1;j<=s;j++){
                if(b[i][j]!='e')a[i][j]=b[i][j];
                b[i][j]='e';
            }
        for(i=1;i<=r;i++)
        for(j=1;j<=s;j++){
            if(a[i][j]=='R') {
                if(a[i-1][j]=='S')b[i-1][j]='R';
                if(a[i+1][j]=='S')b[i+1][j]='R';
                if(a[i][j-1]=='S')b[i][j-1]='R';
                if(a[i][j+1]=='S')b[i][j+1]='R';
            }
            if(a[i][j]=='S') {
                if(a[i-1][j]=='P')b[i-1][j]='S';
                if(a[i+1][j]=='P')b[i+1][j]='S';
                if(a[i][j-1]=='P')b[i][j-1]='S';
```

```
                    if(a[i][j+1]=='P')b[i][j+1]='S';
                }
                if(a[i][j]=='P') {
                    if(a[i-1][j]=='R')b[i-1][j]='P';
                    if(a[i+1][j]=='R')b[i+1][j]='P';
                    if(a[i][j-1]=='R')b[i][j-1]='P';
                    if(a[i][j+1]=='R')b[i][j+1]='P';
                }
            }
        }
        for(i=1;i<=r;i++)
            for(j=1;j<=s;j++)
                if(b[i][j]!='e')a[i][j]=b[i][j];
        for(i=1;i<=r;i++){
            for(j=1;j<=s;j++)
                cout<<a[i][j];
            cout<<endl;
        }
        if(k==t)break;
        cout<<endl;
}
return 0;   }
```

1025、Persistent Numbers

```
#include <iostream>

using namespace std;

int a2,a3,a5,a7;

int a[1001],tem[1001];

char s[1001];

int delive(int n){

	int k=0,i=1,num=0;

	while(a[i]==0 && i<=a[0])i++;

	if(a[i]<n){ k=a[i];    i=i+1; }

	else { k=0;}

	for(;i<=a[0];i++){

		k=k*10+a[i];

		tem[++num]=k/n;

		k=k%n;

	}

	tem[0]=num;

	if(k!=0)return 0;

	for(i=0;i<=num;i++)a[i]=tem[i];

	return 1;

}

int mod(int n){

	int k;
```

```
    if(n==2){
        if(a[a[0]]%2==0){ delive(2);    return 1; }
        return 0;
    }
    if(n==5){
      if(a[a[0]]==0 || a[a[0]]==5){delive(5); return 1;}
        return 0;
    }
    if(n==3){
        k=0;
        for(int i=1;i<=a[0];i++)k+=a[i];
        if(k%3==0){ delive(3);    return 1; }
        return 0;
    }
    if(n==7){
        if(delive(7))return 1;      return 0;
    }
    return 0;
}
int main(){
    int n,k,res,i;
    while(gets(s)){
        if(s[0]=='-')break;
```

```
if(s[1]=='\0'){
        cout<<10+s[0]-'0'<<endl;
        continue;
}
a2=0;a3=0;a5=0;a7=0;
memset(a,0,1001);
memset(tem,0,1001);
for(i=0;s[i]!='\0';i++)a[i+1]=s[i]-'0';
a[0]=i;
while(mod(2)){ ++a2; }
while(mod(3)){ ++a3; }
while(mod(5)){ ++a5; }
while(mod(7)){ ++a7; }
if(a[0]>1){
      cout<<"There is no such number."<<endl;
        continue;
}
k=0;
while(a3>=2){
        a3-=2;          tem[++k]=9;
}
while(a2>=3){
        a2-=3;            tem[++k]=8;
```

```
    }
    while(a7>0){
        --a7;               tem[++k]=7;
    }
    while(a2>0 && a3>0){
        --a2;               --a3;
        tem[++k]=6;
    }
    while(a5>0){
        --a5;               tem[++k]=5;
    }
    while(a2>=2){
        a2-=2;              tem[++k]=4;
    }
    while(a3>0){
        --a3;               tem[++k]=3;
    }
    while(a2>0){
        --a2;               tem[++k]=2;
    }
    while(k>0)cout<<tem[k--];
    cout<<endl;
}
```

```
    return 0;
}
```

1028、Guessing Game

```
#include <stdio.h>
#include <string.h>
int main(){
    int n,i;
    char s[5];
    int jud[11];
    int flag=0;
    for(int j=0;j<11;j++)jud[j]=0;
    while(scanf("%d",&n)){
        if(n==0)break;
        scanf("%s%s",s,s);
        if(s[0]=='o'){
            if(jud[n]!=0)flag=1;
            if(flag==1)printf("Stan is dishonest\n");
            else printf("Stan may be honest\n");
            flag=0;
            for(int j=0;j<11;j++)jud[j]=0;
            continue;
        }
        if(flag==1)continue;
```

```
            if(s[0]=='l'){
                for(i=1;i<=n;i++)
                    if(jud[n]==1){
                        flag=1;     break;
                    }
                for(i=1;i<=n;i++)jud[i]=-1;
                continue;
            }
            if(s[0]=='h'){
                for(i=n;i<=10;i++)
                    if(jud[n]==-1){
                        flag=1;    break;
                    }
                for(i=n;i<=10;i++)jud[i]=1;
            }
        }
        return 0;
}
```

1015、Euclid's Game

```
#include <stdio.h>
int main() {
    int a,b,tem,n;
    while(1){
```

```
        scanf("%d%d",&a,&b);
        if(a==0 && b==0)break;
        if(b>a) {tem=a;a=b;b=tem;}
        if(a/b>1){printf("Stan wins\n");continue;}
        n=0;
        while(1) {
            ++n;            a=a-b;
            tem=a;
            a=b;
            b=tem;
            if(b==0){
                if(n%2==1)printf("Stan wins\n");
                break;
            }
            if(a/b>1) {
                if(n%2==1)printf("Ollie wins\n");
                else printf("Stan wins\n");
                break;
            }
        }
    }
    return 0;
}
```

2108、Elevator(电梯)

```
#include<stdio.h>
int main()
{
    int quest[100];
    int i,j,sum,flag,num;
    scanf("%d",&num);
    for(;num;)
    {
        sum=0;flag=0;
        for(i=0;i<num;i++)
            scanf("%d",quest+i);
        for(j=0;j<num;j++)
        {
            if(quest[j]>flag)
                sum+=(quest[j]-flag)*6+5;
            else
                sum+=(flag-quest[j])*4+5;
            flag=quest[j];
        }
        printf("%d\n",sum);
        scanf("%d",&num);
    }
```

```
}
```

1001、Calculate a + b

```
#include <stdio.h>
int main()
{
    int a,b;
    while(scanf("%d %d",&a, &b) != EOF)
        printf("%d\n",a+b);
    return 0;
}
```

1007、Power Strings（二）（0.33 1208）（字符串权）

```
#include <stdio.h>
#include <string.h>
int main()
{
    char str[1000001];
    int length,power;
    scanf("%s",str);
    while(str[0]!='.' || str[1]!=0)
    {
```

```
        length=strlen(str);
        power=1;
        for(int x=1;x<=length/2;x++)
        {
            if(length%x==0)
            {
                power=length/x;
                for(int i=0;i<length;i+=x)
                {
                    for(int j=0;j<x;j++)
                    {
                        if(str[j]!=str[i+j])
                            goto L1;
                    }
                }
                goto L2;
L1:             power=1;
            }
        }
L2:     printf("%d\n",power);
        scanf("%s",str);
}
return 0;
```

```
}
```

1007、Power Strings（三）(0.14　1208)（字符串权）

```
#include <stdio.h>
#include <string.h>
char str[1000000];
int i,flags,len,CmpLen;
int main()
{
    scanf("%s",str);    len=strlen(str);
    while(str[0]!='.' || str[1]!=0)
    {
        CmpLen=1;
        for(i=1;i<=len/2;i++)
        {
            CmpLen=i;
            if (len%CmpLen==0)
            {
                flags=0;
                for(int j=1;j<len/CmpLen;j++)
               {
                    for(int c=0;c<CmpLen;c++)
                     {
                        if(str[CmpLen*j+c]==str[c])
```

```
                    {   continue;           }
                else                        {   CmpLen=CmpLen*j+c+1;
                                                flags=1;
                    break;
                }
            }
            if (flags)
            {
                i=CmpLen-1;
                break;
            }
        }
        if (!flags) break;
    }
    else
        continue;
}
printf("%d\n",len/i);
scanf("%s",str);
len=strlen(str);
}
}
```

1813、Biker's Trip Odometer（自行车里程）

```
#include <stdio.h>

int main()

{

    float d,t,s;

    int r,n=0;

    scanf("%f%d%f",&d,&r,&t);

    while(r!=0)

    {

        n++;

        s=d*3.1415927*r/12/5280;

        printf("Trip #%d: %.2f %.2f\n",n,s, s*3600/t);

        scanf("%f%d%f",&d,&r,&t);

    }

}
```

1814、Candy Sharing Game（一）（分糖）

```
#include <iostream>

#include <algorithm>

using namespace std;
```

```
int main()
{
    int n;
    int counts[1000];
    int tmp[1000];
    int i;
    int rounds, total;
    while(1)
    {
        cin >> n;
        if(0 == n)
            break;
        rounds = total = 0;
        for(i = 0; i < n; i++)
            cin >> counts[i];

        while(1)
        {
            copy(counts, counts + n, tmp);
            for(i = 1; i < n; i++)
            {
                counts[i] -= counts[i] / 2;
                counts[i] += tmp[i - 1] / 2;
```

```
                if(counts[i] % 2 == 1)
                        counts[i]++;
            }
            counts[0] -= counts[0] / 2;
            counts[0] += tmp[n - 1] / 2;
            if(counts[0] % 2 == 1)
                        counts[0]++;
            int temp = counts[0];
            i = 1;
            while(i < n && counts[i] == counts[0] )
                i++;
            total++;
            if(i == n)
                break;

        }
        cout << total << ' ' << counts[0] << endl;
    }
    return 0;
}
```

1970 All in All （总的说来）

```
#include<stdio.h>
#include<string.h>
```

```
char	a[100000],b[100000];
int	l[2000000],i,j,m,n;
int	main()
{
while(scanf("%s	%s",a,b)!=EOF)
{m=strlen(a);
	n=strlen(b);
	memset(l,0,sizeof(l));
	for(i=1;i<=m;i++)
for(j=1;j<=n;j++)
if(a[i-1]==b[j-1])
l[i*n+j]=l[(i-1)*n+j-1]+1;
else
l[i*n+j]=l[(i-1)*m+j]>l[i*n+j-1]?l[(i-1)*n+j]:l[i*n+j-1];
	if(l[m*n+n]==m)	printf("Yes\n");
		else	printf("No\n");
}
return	0;
}
```

1733、 Common Subsequence（一般后果）

```
#include<stdio.h>
#include<string.h>
int main(){
char a[1000],b[1000];
        int l[1000][1000],i,j,m,n;
 while(scanf("%s %s",a,b)!=EOF)
 {m=strlen(a);
  n=strlen(b);
  for(i=1;i<=m;i++)
  for(j=1;j<=n;j++)
    if(a[i-1]==b[j-1])
      l[i][j]=l[i-1][j-1]+1;
    else
      l[i][j]=l[i-1][j]>l[i][j-1]?l[i-1][j]:l[i][j-1];
 printf("%d\n",l[m][n]);
 }
 return 0;
```

```
}
```

2481、Unique Ascending Array（递增数组）

```
#include <stdio.h>
int main()
{
  void sort(int *p,int iCount);
  int iCount,i,iNum[100]={0};
  do
  {
    scanf("%d",&iCount);
    for(i=0;i<iCount;i++)   {
      scanf("%d",&iNum[i]);
    }
    sort(iNum,iCount);
    for(i=0;i<iCount;i++)   {
      if(i==iCount-1)
      {
          printf("%d\n",iNum[i]); }
      else if(iNum[i]!=iNum[i+1])
      {
          printf("%d ",iNum[i]);
      }
    }
```

```
  }while(iCount);
  return 0;
}
void sort(int *p,int iCount)
{
  int temp,i,j;
  for(i=0;i<iCount;i++) {
    for(j=0;j<i;j++)    {
      if(p[i]<p[j])    {
        temp=p[i];
        p[i]=p[j];
        p[j]=temp;
      }
    }
  }
}
```

2478、Encoding（编码）

```
#include <stdio.h>
using namespace std;
int main()
{
        int iLine,i,j,iNum;
        char sStr[100];
```

```
scanf("%d",&iLine);
for(i=0;i<iLine;i++)
{
    scanf("%s",&sStr);
    j=0;
    iNum=1;
    for(j=0;j<100 && sStr[j]!=0;j++)
    {
        if(sStr[j]==sStr[j+1])
            iNum++;
        else
        {
                if(iNum==1)
                    printf("%c",sStr[j]);
                else
                {
                    printf("%d%c",iNum,sStr[j]);
                    iNum=1;
                }
        }
    }
    printf("\n");
}
```

```
    return 0;
}
```

2417、Lowest Bit(最小位)

```
#include <stdio.h>
int main()
{
   int iNum,iArit,i,j,iSum;
   while(scanf("%d",&iNum))
   {
       if(iNum==0) break;
       i=0;
       do
       {
           i++;
           iArit=iNum%2;
           iNum=iNum/2;
       }while(iArit==0);
       iSum=1;
       for(j=1;j<i;j++)
```

```
        {
            iSum=iSum*2;
        }
        printf("%d\n",iSum);
    }
    return 0;
}
```

1962、How Many Fibs?(一) （0.01 444）(斐波那契)

```
#define MAXLENGTH1 481
#define MAXLENGTH2 105
#include <stdio.h>
int main()
{
    char a[MAXLENGTH1][MAXLENGTH2],b1[MAXLENGTH2],b2[MAXLENGTH2],*pe,*ph,temp;
    char *pb1,*pb2;
    int i,j,lmax,jwei,tep,lb1,lb2,head,end,index,db1,db2;
    a[0][0]=1;a[0][1]=0;a[0][MAXLENGTH2-1]=1;
    a[1][0]=2;a[1][1]=0;a[1][MAXLENGTH2-1]=1;
    i=2;lmax=1;
    while (i<MAXLENGTH1)
    {
        jwei=0;tep=0;
        for (j=0;j<lmax;j++)
```

```
        {
            tep=jwei+a[i-1][j]+a[i-2][j];
            a[i][j]=tep%10;
            jwei=tep/10;
        }
        while (jwei>0)
        {
            a[i][j++]=jwei%10;
            jwei=jwei/10;
        }
        a[i][MAXLENGTH2-1]=j;
        a[i][j]=0;
        lmax=j;
        i++;
}
scanf("%s %s",b1,b2);
while (b1[0]!='0'||b1[1]!=0||b2[0]!='0'||b2[1]!=0)
{
        if (b1[0]=='0'&&b1[1]==0) b1[0]=49;
        pb1=b1;lb1=0;
        pb2=b2;lb2=0;
        while (*pb1!=0)
        {
```

```
    lb1++;
    *pb1-=48;
    pb1++;
}
b1[MAXLENGTH2-1]=lb1;
while (*pb2!=0)
{
     lb2++;
     *pb2-=48;
     pb2++;
}
b2[MAXLENGTH2-1]=lb2;
pe=&b1[lb1-1];ph=b1;
while (ph<pe)
{
    temp=*ph;
    *ph=*pe;
    *pe=temp;
    ph++;
    pe--;
}
pe=&b2[lb2-1];ph=b2;
while (ph<pe)
```

```
 {
      temp=*ph;
      *ph=*pe;
      *pe=temp;
      ph++;
      pe--;
}
head=0;end=MAXLENGTH1-1;int tx=0;
while (1)
{
    index=(head+end)/2;
    if (lb1>a[index][MAXLENGTH2-1])    head=index;
    else if(lb1<a[index][MAXLENGTH2-1]) end=index;
           else
         {
             for (i=lb1-1;i>=0;i--)
             {
                 if (b1[i]>a[index][i]) {head=index;break;}
                 else if(b1[i]<a[index][i]) {end=index;break;}
           }
         }
    if (head!=index&&end!=index) {db1=index;break;}
    if (index==head&&index==end-1)       {tx++;}
```

```
        if (index==head&&index==end-1&&tx==2)       {db1=end;break;}
}
if (b1[0]==1&&b1[1]==0&&b1[MAXLENGTH2-1]==1) db1=0;
head=0;end=MAXLENGTH1-1;
while (1)
{
        index=(head+end)/2;
        if (index==head&&index==end-1)       {db2=head;break;}
        if (lb2>a[index][MAXLENGTH2-1])    head=index;
        else if(lb2<a[index][MAXLENGTH2-1]) end=index;
                else
                {
                        for (i=lb2-1;i>=0;i--)
                        {
                                if (b2[i]>a[index][i]) {head=index;break;}
                                else if(b2[i]<a[index][i]) {end=index;break;}
                        }
                }
                if (head!=index&&end!=index) {db2=index;break;}
          }
      if (head==end)    {db2=head;}
      i=db2-db1+1;
      printf("%d\n",i);
```

```
            scanf("%s %s",b1,b2);
        }
    return 1;
}
```

2176、Speed Limit（速度限制）（一）　（0　388）

```
#include<stdio.h>
int main()
{
    int n,s[10],t[10],total[10]={0};
    int i,k=0,temp=0;
    while(scanf("%d",&n))
    {
        if(n==-1)break;
        temp=0;
        for(i=0;i<n;i++)
        {
            scanf("%d%d",&s[i],&t[i]);
            total[k]+=s[i]*(t[i]-temp);
            temp=t[i];
        }
        k++;
    }
    for(i=0;i<k;i++)
```

```
    {
        printf("%d miles\n",total[i]);
    }
    return 0;
}
```

2176、Speed Limit（速度限制）（二）　(0　388)

```
#include<stdio.h>
int main()
{
    int j,n,l,w, y=0, k=11;
    int s[10]={0,0,0,0,0,0,0,0,0,0};
    int t[11]={0,0,0,0,0,0,0,0,0,0};
    int temp[10];
    for(int x=0;x<=10;x++) {
```

```
        temp[x]=0;
    }
    int flag=1;
f1: while(flag)
    {
f2:     scanf("%d",&n);
        w=j=l=n;
        while(n>0)
        {
            scanf("%d%d",&s[n],&t[n]);
            n--;
        }
        k--;
        if(n>=0){
            while(k)
            {
                for(l;l>0;l--)
                {
                    temp[k]+=t[l]*s[l]-y*s[l];
                    y=t[l];
                }
                y=0;
                goto f2;
```

```
                }
            }
            if(w==-1)
            {
                for(int p=10;p>=0;p--)
                {
                    if(temp[p]!=0)
                        printf("%d miles\n",temp[p]);
                }
                flag=0;
                goto f1;
            }
        }
        return 0;
}
```

2104、Let The Balloon Rise（气球）

```
#include<stdio.h>
#include<string.h>
struct ballon
{
    char color[16];
    int number;
};
```

```
int main()
{
    int n,i,j,k,flag,max,maxnum;
    struct ballon bal[1000];
    char input[16];
    while(scanf("%d",&n)&&n!=0)
    {
        k=0;max=0;
        for(i=0;i<n;i++)
        {
            scanf("%s",&input);
            flag=0;
            for(j=0;j<k;j++)
            {
                if(strcmp(input,bal[j].color)==0)
                {
                    bal[j].number++;
                    flag=1;
                }
            }
            if (flag==0)
            {
                strcpy(bal[k].color,input);
```

```
                bal[k].number=1;
                k++;
            }
        }
        for(i=0;i<k;i++)
            if(max<bal[i].number)
            {
                max=bal[i].number;
                maxnum=i;
            }
        printf("%s\n",bal[maxnum].color);
    }
    return 0;
}
```

1962、How Many Fibs?(二) (0.07　396)(斐波那契)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAXLEN 101
 int max(int a ,int b)
 {
    return a>=b?a:b;
```

```
}

void add(const int len_a, const char a[], const int len_b, const char b[], int *len_c, char c[])

{

    int i, j, temp, carry = 0;

    *len_c = 0;

    j = MAXLEN - max(len_a, len_b);

    for (i = MAXLEN - 1; i >= j; --i)

    {

        temp = a[i] + b[i] + carry;

        c[i] = temp % 10;

        carry = temp / 10;

        ++(*len_c);

    }

    if (carry)

    {

        c[i] = carry;

        ++(*len_c);

    }

}

 void init(const int len, char a[], int is_char)

{

    int i, j = MAXLEN - 1;

    for (i = len - 1; i >= 0; --i)
```

```
        a[j--] = a[i] - (is_char ? '0' : 0);
}
int compare(const int len_a, const char a[], const int len_b, const char b[])
{
    int i;
    if (len_a < len_b)
        return -1;
    else if (len_a > len_b)
        return 1;
    for (i = MAXLEN - len_a; i < MAXLEN; ++i)
    {
        if (a[i] < b[i])
            return -1;
        else if (a[i] > b[i])
            return 1;
    }
    return 0;
}
int main()
{
    char a[MAXLEN], b[MAXLEN], fn1[MAXLEN], fn2[MAXLEN], fib[MAXLEN];
    int len_fn1, len_fn2, len_fib, len_a, len_b, result1, result2, count;
    while (1)
```

```
{
    scanf("%s %s", a, b);
    if ('0' == a[0] && '\0' == a[1] && '0' == b[0] && '\0' == b[1])
        break;
    len_a = strlen(a);
    len_b = strlen(b);
    init(len_a, a, 1);
    init(len_b, b, 1);
    count = 0;
    memset(fn1, 0, MAXLEN);
    memset(fn2, 0, MAXLEN);
    fn1[0] = 0;
    fn2[0] = 1;
    len_fn1 = 1;
    len_fn2 = 1;
    init(len_fn1, fn1, 0);
    init(len_fn2, fn2, 0);
    memset(fib, 0, MAXLEN);
     while (1)
    {
     add(len_fn1, fn1, len_fn2, fn2, &len_fib, fib);
        result1 = compare(len_fib, fib, len_a, a);
        result2 = compare(len_fib, fib, len_b, b);
```

```
                if (result2 > 0)
                    break;
                if (0 <= result1 && result2 <= 0)
                    ++count;
                len_fn1 = len_fn2;
                memcpy(fn1, fn2, MAXLEN);
                len_fn2 = len_fib;
                memcpy(fn2, fib, MAXLEN);
            }
            printf("%d\n", count);
        }
    }
```

1789、The Suspects（疑心）

```
#include<iostream>
#include<cstdio>
using namespace std;
#define N 40000
int r[N];
int findr(int a)
{
while(r[a]!=a)a=r[a];
return a;
}
```

```
int      main()

{

     int      n,m;

     while(scanf("%d      %d",&n,&m)==2)

     {

        if(!n      &&      !m)break;

        int      i,j,k,t,t1;

        for(i=0;i<n;i++)r[i]=i;

        for(i=0;i<m;i++)

        {

          scanf("%d",&k);

          scanf("%d",&t);

          t=findr(t);

          for(j=1;j<k;j++)

          {

              scanf("%d",&t1);

              if(findr(t1)!=t)

              {

                   r[findr(t1)]=t;

              }

          }

       }

       int      ans;
```

```
        t=findr(0);
        for(i=1,ans=1;i<n;i++)
            if(findr(i)==t)ans++;
        printf("%d\n",ans);
    }
  return    0;
}
```

1890、Average Speed(平均速度)

```
#include <iostream>
#include <iomanip>
#include <stdio.h>
using namespace std;
int main()
{
    double h, m , s, speed = 0 , h1 = 0,m1= 0 ,s1 = 0,time = 0,temp1,temp2,temp3 ,flag1,flag2;
    char temp;
    double total=0 ;
    while (cin>>h )
    {
                cin.get();
                cin>>m;
                cin.get();
                cin>>s;
```

```
        temp3 = s - s1;
        flag1 = flag2 = 0;
        if ( temp3 <0 )
        {
                flag1 = 1;
                temp3 = 60 + temp3;
        }
        temp1 = m - m1 ;
        if (flag1 ) temp1--;
        if ( temp1< 0 )
        {
                flag2 = 1;
                temp1 = 60 + temp1;
        }
        temp2 = h - h1;
        if (flag2 ) temp2--;
        total = (temp3 /3600 + temp1/60 + temp2 ) * speed* 1.0 +total;
        s1 = s;         m1 = m;h1 =h;
cin.get (temp );
if ( temp!='\n')
        cin>>speed;
else
{       cout<<fixed<<setprecision(0);
```

```
                if ( h < 10 )
                        cout<<'0'<<h<<':';
                if ( m < 10 )
                        cout<<'0';
                cout<<m<<':';
                if ( s < 10 )
                        cout<<'0';
                cout<<s<<' '<<fixed<<setprecision(2)<<total <<" km"<<endl;
            }
        }
        return 0; }
```

2060、Fibonacci Again(斐波那契)

```
#include      <iostream.h>
#include<stdio.h>
using      namespace      std;
char      r[1000000];
int      main()
{
   int      a=1,b=2,c;
   for(int      i=2;i<1000000;i++)
   {
            c=(a+b)%3;
            if(!c)
```

```
                r[i]=1;
            a=b;
            b=c;
    }
    int     n;
    while(cin>>n)
    {
        if(r[n])
            cout<<"yes"<<endl;
        else
        cout<<"no"<<endl;
  }
  return    0;
}
```

2191、Series Determination（方程 $f(x)=AX^2+BX+C$）

```
#include <iostream>
using namespace std;
int main()
{
    int f0, f1 , f2, a, b , c ,temp1,temp2 , i,j ,f3, f4 ,f5;
      while ( cin>>f0>>f1>>f2)
      {
            c = f0;
```

```
            temp1 = f1 -c; temp2 = f2 - c;
            b = 2*temp1 - temp2 /2;
            a = temp2/2 - temp1;
            f3 = 9 * a + 3 * b + c;
            f4 = 16 * a + 4 * b + c;
            f5 = 25 *a + 5 * b + c;
            cout<<f3<<' '<<f4<<' '<<f5<<endl;
        }
        return 0;
}
```

1814、Candy Sharing Game（二）（分糖）

```
#include <stdio.h>
int main()
{
        int sweet[100],buffer[100];
        int round=0,i,child,seat;
        scanf("%d",&child);
        for(i=0;i<child;i++)
                scanf("%d",&sweet[i]);
        do
        {
                round=0;
```

```
while(true)
{

    for(i=0;i<child;i++)
        if(sweet[i]!=sweet[0]) break;
    if(i==child) break;

    for(i=0;i<child;i++)
      sweet[i]=sweet[i]/2,buffer[i]=sweet[i];
    for(i=0;i<child;i++)
    {
        seat=i-1;
        if(seat<0) seat=child-1;
        sweet[i]+=buffer[seat];
    }
    for(i=0;i<child;i++)
            if(sweet[i]%2!=0) sweet[i]++;
    round++;
}
printf("%d %d\n",round,sweet[0]);
scanf("%d",&child);
for(i=0;i<child;i++)
    scanf("%d",&sweet[i]);
```

```
        }while(child!=0);

        return 0;

    }
```

1657、Goldbach's Conjecture（歌德拔河）

```
#include <iostream.h>

int isnprime[32768];

int prime[3600];

int pl;

void init()

{

        int i,j;

        for(i=2;i<32768;i++)

        {

                if(isnprime[i])
```

```
                continue;
            prime[pl++]=i;
            for(j=i*i;j<32768;j+=i)
                isnprime[j]=1;
        }
}
int go(int n){
        int i,r=0;
        for(i=0;i<pl && prime[i]<=n/2;i++){
            if(!isnprime[n-prime[i]])
            r++;
        }
        return r;
}
int main()
{
        int n;
        init();
        while(cin>>n && n){
            cout<<go(n)<<endl;
        }
        return 0;
}
```

1828、Fibonacci Numbers（斐波那契数列）

```
#include<stdio.h>
#include<string.h>
const int bitlong=1100;
struct Bigint{
 int len;      135408167271
 char bit[bitlong];
};
Bigint add(Bigint a,Bigint b){
 Bigint c;
 int i;
 if(a.len<b.len) {
  for(i=a.len;i<b.len;i++)
    a.bit[i]=0;
  c.len=b.len;
 }
 else {
  for(i=b.len;i<a.len;i++)
    b.bit[i]=0;
  c.len=a.len;
 }
 int k=0;
 for (i=0;i<c.len;i++) {
```

```
    c.bit[i]=a.bit[i]+b.bit[i]+k;
    if(c.bit[i]>9) {
      k=1;
      c.bit[i]-=10;
    }
    else
      k=0;
  }
  if(k==1) {
    c.bit[c.len]=k;
    c.len++;
  }
  return c;
}
void print(Bigint a){
  int i;
  i=a.len-1;
  while(i>-1)
    printf("%d",a.bit[i--]);
  printf("\n");
}
Bigint s[5000];
int main(){
```

```
  int i,n;
  s[1].bit[0]=1;
  s[2].bit[0]=1;
  s[1].len=1;
  s[2].len=1;
  for (i=3;i<5000;i++) {
    s[i]=add(s[i-1],s[i-2]);
  }
  while(scanf("%d",&n)!=EOF) {
    print(s[n]);
  }
  return 0;
}
```

2001、Adding Reversed Numbers（倒序相加）

```
#include <iostream>
#include <string.h>
using namespace std;
void rever();
void sum();
char str1 [ 1000 ] ,str2 [ 1000 ];
int num1 [ 1000 ],num2 [ 1000 ],num[ 1000 ] , len1,len2 , len;
int main(){
```

```
        int i , cas;
        cin>>cas;
        while ( cas){
                cin>>str1;
                cin>>str2;
                len1 = strlen(str1 );
                len2 = strlen (str2 );
                rever();
                sum();
                i = len-1;
                while ( num [ i ] ==0 )
                        i --;
                for ( ; i >=0 ; i-- )
                        cout<<num [ i ];
                cout<<endl;
                cas--;
        }
        return 0;
}
void rever(){
        int i;
        char tc;
        for ( i = 0 ; i <len1/2 ; i ++ ){
```

```
            tc = str1 [ i ];
            str1 [ i ] = str1 [ len1 - i-1 ] ;
            str1[ len1 - i-1 ] = tc;
        }
        for ( i = 0 ; i<len2/2; i ++ ){
            tc = str2 [ i ];
            str2 [ i ] = str2 [ len2 - i -1 ] ;
            str2 [ len2 - i-1 ] = tc;
        }
}
void sum(){
        int i , j , tmp, lent , carry ;
        for ( i = 0 ; i<len1 ; i ++ )
            num1 [ i ] = str1 [ i ] - '0';
        for ( i = 0 ; i<len2 ; i ++ )
            num2 [ i ] = str2 [ i ] - '0';
        if ( len1>len2 ){
            len = len1;
            lent = len2;
        }
        else{
            len = len2;
            lent = len1;
```

```
}
carry = 0;
for ( i = 1 ; i <=lent ; i ++ ){
        num [ len - i ] = num1 [ len1 - i ] + num2 [ len2 - i ] +carry ;
    carry = 0;
    while ( num [ len - i ] >=10 ) {
        carry ++;
        num [ len - i ] -= 10;
    }
}
if ( len1 >len2 )
    for ( ; i <=len1 ; i ++ ){
        num [ len - i ] = num1 [ len1- i ] +carry ;
        carry = 0;
        while ( num [ len - i ] >=10 ) {
            carry ++;
            num [ len - i ] -=10;
        }
    }
else
    for ( ; i<=len2 ; i ++ ){
        num [ len-i ] = num2 [ len2-i ] + carry ;
        carry = 0;
```

```
                while ( num [ len- i ] >= 10 ) {
                        carry ++;
                        num [ len - i ] -=10;
                }
        }
if ( carry >0 ) {
                for ( i = len ; i >0 ; i -- )
                        num [ i ] = num [ i-1 ];
                num [ 0 ] =carry ;
                len ++;
        }
}
```

1976、Paths on a Grid

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{
        double m , n ,sum ,temp ;
        long i;
        while (cin>> m >> n && ( m||n ) )
```

```
        {
                sum = 1.0;
                        if ( n<m )
                {
                        temp = m;
                        m = n;
                        n = temp;
                }
                for ( i=1; i<=m ; i++ )
                        sum*=double(n+ i)/i;
            cout<<fixed<<setprecision(0)<<sum<<endl;
        }
        return 0;
}
```

1229、Gift?!

```
#include <stdio.h>
#include <string.h>
int m, n, g;
int a[49];
int main(){
  //freopen("in.txt", "r", stdin);
  while(scanf("%d %d", &n, &m) && m | n) {
    g = 0;
```

```
memset(a, 0, sizeof(a));
a[0] = 1;
if(n < 50) {
 int got = 1;
 int c = 2;
 while(got && !a[m - 1]) {
  got = 0;
  for(int i = 0; i < n; i ++){
   if(a[i] == c - 1){
    int i0 = i + 2 * c - 1;
    int i1 = i - 2 * c + 1;
    if(i0 < n){
     got = 1;
     a[i0] = c;
    }
    if(i1 >= 0) {
     got = 1;
     a[i1] = c;
    }
   }
  }
  c ++;
 }
```

```
    if(a[m - 1])
      g = 1;
   }
   else
    g = 1;
   if(g)
    printf("Let me try!\n");
   else
    printf("Don't make fun of me!\n");
 }
 return 0;
}
```

1119、SPF

```
#include <cstdio>
#include <string>
int x, y, vx, idx, a[1001][1001], dfn[1001], low[1001], sub[1001], root, c = 0, mx, mn;
// low[] means the lowest point one root and its subtree point to
int min(int a, int b)
{
  return a > b ? b : a;
}
void init()
{
  memset(dfn, 0, sizeof(dfn));
  memset(low, 0, sizeof(low));
  memset(sub, 0, sizeof(sub));
  idx = 1, low[vx] = dfn[vx] = idx, root = 0;
}
void dfs(int i)
{
  int k;
  for(k = mn; k <= mx; k ++)
  {
    if(a[i][k])    // k adjacent to i
```

```
    {
      if(!dfn[k])  // k not visited -- k is in subtree
      {
        idx ++;
        dfn[k] = low[k] = idx;
        dfs(k);
        low[i] = min(low[i], low[k]);
        // and if k is subtree, judge if it is AP(articulation point)
        if(low[k] >= dfn[i])
        {
          if(i == vx) root ++;
          if(i != vx)  sub[i] ++;
        }
      }
      else // k visited -- k is ancestor
      {
        low[i] = min(low[i], dfn[k]);
      }
    }
  }
}
void print()
{
  if(c ++) printf("\n");
  printf("Network #%d\n", c);
  int i, f = 0;
  //pl();
  if(root > 1) sub[vx] = root - 1;
  for(i = mn; i <= mx; i ++)
  {
    if(sub[i])
    {
      f = 1;
      printf("   SPF node %d leaves %d subnets\n", i, sub[i] + 1);
    }
  }
  if(!f)
    printf("   No SPF nodes\n");
}

int main()
{
  //freopen("in.txt", "r", stdin);
  while(scanf("%d", &x) && x)
  {
    vx = x;
    memset(a, 0, sizeof(a));
```

```
    scanf("%d", &y);
    a[x][y] = 1;
    a[y][x] = 1;
    mx = x, mn = x;
    if(y > mx) mx = y;
    if(y < mn) mn = y;
    while(scanf("%d", &x) && x)
    {
      scanf("%d", &y);
      a[x][y] = 1;
      a[y][x] = 1;
      if(x > mx) mx = x;
      if(x < mn) mn = x;
      if(y > mx) mx = y;
      if(y < mn) mn = y;
    }
    init();
    dfs(vx);
    print();
  }
  return 0;
}
```

1268、Is It A Tree?

```
#include <cstdio>

#include <string>

int p[1001], x, y, b[1001], ac, c[1001], k = 1;

int find_set(int i)

{

 if(p[i] != i)

 {

  p[i] = find_set(p[i]);

 }

 return p[i];

}
```

```
void proc()
{
 int i, t = 0;
 for(i = 0; i < 1001; i ++)
 {
  if(b[i])
  {
   c[find_set(i)] ++;
  }
 }
 for(i = 0; i < 1001; i ++)
 {
  if(c[i] > 1)
   t ++;
 }
 if(t > 1) ac = 0;
}
  void init()
{
 int i;
 for(i = 0; i <= 1000; i ++)
  p[i] = i;
 ac = 1;
```

```
 memset(b, 0, sizeof(b));
 memset(c, 0, sizeof(c));
}
void pt()
{
 printf("Case %d is ", k ++);
 if(ac)
  printf("a tree.\n");
 else
  printf("not a tree.\n");
}
int main()
{
 //freopen("in.txt", "r", stdin);
 init();
 while(scanf("%d %d", &x, &y))
 {
  if(x == -1)
   break;
  if(x == 0)
  {
   proc();
   pt();
```

```
    init();
    continue;
   }
   b[x] = 1, b[y] = 1;
   if(p[y] != y || find_set(x) == y) ac = 0;
   p[y] = find_set(x);
  }
  return 0;
}
```

1234、Chopsticks

```
#include <cstdio>
#include <string>
#define MX 99999999
int a[5001], b[2][5001], c, n, m;
void init()
{
 memset(b, 0, sizeof(b));
}
void dp()
{
 int i, k;
 for(i = 1; i <= m; i ++)
 {
```

```
  for(k = 2 * i; k <= n; k ++)
  {
   b[1][k] = MX;
   if(k > 2 * i) b[1][k] = b[1][k - 1];
   if(n - k > (m - i) * 3)
   {
    int t = b[0][k - 2] + (a[k] - a[k - 1]) * (a[k] - a[k - 1]);
    if(t < b[1][k]) b[1][k] = t;
   }
  }
  memcpy(b[0], b[1], sizeof(b[1]));
 }
}
void prt()
{
 //pt();
 printf("%d\n", b[1][n]);
}
int main()
{
 //freopen("in.txt", "r", stdin);
 scanf("%d", &c);
 int i, k;
```

```
for(i = 0; i < c; i ++)
{
 scanf("%d %d", &m, &n);
 m += 8;
 for(k = 1; k <= n; k ++)
 {
  scanf("%d", &a[k]);
 }
 init();
 dp();
 prt();
}
return 0;
}
```

1084、Channel Allocation

```
#include <cstdio>
#include <string>
int a[26][26], n, m, ans, c[26], b[26];
char s[26];
void greedy()
{
  int i, k;
  for(i = 0; i < n; i ++)
  {
    memset(b, 0, sizeof(b));
    for(k = 0; k < n; k ++)
    {
      if(a[i][k] && c[k] != -1)
      {
        b[c[k]] = 1;
```

```
      }
    }
    for(k = 0; k <= i; k ++)
    {
      if(!b[k]) break;
    }
    c[i] = k;
  }
  for(i = 0; i < n; i ++)
  {
    if(ans < c[i])
      ans = c[i];
  }
  ans ++;
}
void init()
{
  memset(a, 0, sizeof(a));
  for(int i = 0; i < n; i ++)
  {
    c[i] = -1;
  }
  ans = 0;
}
int main()
{
  //freopen("in.txt", "r", stdin);
  while(scanf("%d", &n) && n)
  {
    int i, k;
    init();
    for(i = 0; i < n; i ++)
    {
      scanf("%s", &s);
      m = strlen(s) - 2;
      for(k = 0; k < m; k ++)
      {
        a[i][s[k + 2] - 'A'] = 1;
        a[s[k + 2] - 'A'][i] = 1;
      }
    }
    greedy();
    if(ans != 1)
      printf("%d channels needed.\n", ans);
    else
      printf("%d channel needed.\n", ans);
  }
```

```
  return 0;
}
```

1789、The Suspects

```
#include <cstdio>

#include <string>

int p[30001], n, m, t, x, y, ans;

int findset(int i)

{

 if(p[i] != i)

  p[i] = findset(p[i]);

 return p[i];

}

void init()

{

 int i;

 for(i = 0; i < n; i ++)

 {

  p[i] = i;

 }

}

int main()

{

 //freopen("in.txt", "r", stdin);

 while(scanf("%d %d", &n, &m) && n)
```

```
{
 int i, k;
 init();
 for(i = 0; i < m; i ++)
 {
  scanf("%d", &t);
  scanf("%d", &x);
  x = findset(x);
  for(k = 1; k < t; k ++)
  {
   scanf("%d", &y);
   if(x != findset(y))
    p[findset(y)] = x;
  }
 }
 ans = 1;
 for(i = 1; i < n; i ++)
 {
  if(findset(i) == findset(0))
   ans ++;
 }
 //pt();
 printf("%d\n", ans);
```

```
 }
 return 0;
}
```

1141、Closest Common Ancestors

```
#include <cstdio>
int a[20] = { 1, 1, 2, 5, 14, 42, 132,
  429, 1430, 4862, 16796, 58786, 208012,
  742900, 2674440, 9694845, 35357670,
  129644790, 477638700 }, b[20], n;
void dfs(int x)
{
 int s, l, r, i, t = x, xl, xr;
 for(i = 0; i < 20; i ++)
 {
  if(t < a[i])
   break;
  t -= a[i];
 }
 s = i - 1;
 for(i = 0; i <= s; i ++)
 {
  if(t < a[i] * a[s - i])
   break;
```

```
  t -= a[i] * a[s - i];
 }
 l = i, r = s - i;
 xl = t / a[r] + b[l], xr = t % a[r] + b[r];
 if(xl)
 {
  printf("(");
  dfs(xl);
  printf(")");
 }
 printf("X");
 if(xr)
 {
  printf("(");
  dfs(xr);
  printf(")");
 }
}

void init()
{
 int i, t = 0;
 b[0] = 0;
```

```
 for(i = 0; i < 19; i ++)

 {

  t += a[i];

  b[i + 1] = t;

 }

}
int main()
{
  //freopen("in.txt", "r", stdin);
  init();
  while(scanf("%d", &n) && n)
  {
    dfs(n);
    printf("\n");
  }
  return 0;
}
```

1425、Crossed Matchings

```
#include <cstdio>

#include <string>

int n, an, bn, a[101], b[101], r[101][101];

void dp()

{

 int i, k, mx, j, u;

 memset(r, 0, sizeof(r));

 for(i = 1; i <= an; i ++)

 {

  for(k = 1; k <= bn; k ++)

  {
```

```
mx = 0;
if(a[i] != b[k])
{
 for(j = i - 1; j > 0; j --)
 {
  if(a[j] == b[k])
  {
   for(u = k - 1; u > 0; u --)
   {
    if(b[u] == a[i])
    {
     if(u && j && mx < r[j - 1][u - 1] + 2)
     {
      mx = r[j - 1][u - 1] + 2;
     }
    }
   }
  }
 }
}
if(mx < r[i - 1][k])
 mx = r[i - 1][k];
if(mx < r[i][k - 1])
```

```
      mx = r[i][k - 1];
    if(mx < r[i - 1][k - 1])
      mx = r[i - 1][k - 1];
    r[i][k] = mx;
  }
 }
}
void print()
{
 //pr();
 printf("%d\n", r[an][bn]);
}

int main()
{
 //freopen("in.txt", "r", stdin);
 scanf("%d", &n);
 int i, k;
 for(i = 0; i < n; i ++)
 {
  scanf("%d %d", &an, &bn);
  for(k = 1; k <= an; k ++)
  {
```

```
    scanf("%d", &a[k]);
  }
  for(k = 1; k <= bn; k ++)
  {
    scanf("%d", &b[k]);
  }
  dp();
  print();
 }
 return 0;
}
```

1141、Closest Common Ancestors

```
#include <cstdio>
#include <string>
int n, m, p[1001][2], ans[1001];
int cca(int a, int b)
{
 while(p[a][1] > p[b][1])
 {
  a = p[a][0];
 }
 while(p[b][1] > p[a][1])
 {
```

```
  b = p[b][0];
 }
 while(p[a][1])
 {
  if(a == b)
   break;
  a = p[a][0], b = p[b][0];
 }
 return a;
}
int main()
{
 //freopen("in.txt", "r", stdin);
 int i, j, x, y;
 char c;
 while(scanf ("%d", &n) == 1)
 {
  memset(ans, 0, sizeof(ans));
  memset(p, 0, sizeof(p));
  for (i = 0; i < n; i ++)
  {
   scanf ("%d", &x);
   scanf ("%c", &c);
```

```
 scanf ("%c", &c);
 m = 0;
 while (1)
 {
  scanf ("%c",&c);
  if (c >= '0' && c <= '9')
   m = m * 10 + (int)c - 48;
  else break;
 }
 scanf ("%c",&c);
 for(j = 0; j < m; j ++)
 {
  scanf("%d", &y);
  p[y][0] = x;
  p[y][1] = p[x][1] + 1;
 }
}
scanf ("%d\n", &m);
for (i = 0; i < m; i ++)
{
 scanf ("%c", &c);
 x = 0, y = 0;
 while (1)
```

```
 {
  scanf ("%c", &c);
  if ((c >= '0') && (c <= '9'))
   x = x * 10 + (int)c - 48;
  else break;
 }
 while (1)
 {
  scanf ("%c", &c);
  if ((c >= '0') && (c <= '9'))
   y = y * 10 + (int)c - 48;
  else break;
 }
 scanf ("%c", &c);
 //printf ("%d %d\n", x, y);
 ans[cca (x, y)] ++;
}
for(i = 1; i <= n; i ++)
{
 if(ans[i])
  printf ("%d:%d\n", i, ans[i]);
 }
}
```

```
 return 0;
}
```

1180、Self Numbers

```
#include<iostream>
using namespace std;
int b[1500001];
int i;
int opot(int m)
{
 int n=m;
 int tp=0;
 while(1)
 {
  if(n/10==0)
    break;
  else
  {
    tp+=n%10;
    n/=10;
  }
 }
 tp+=n;
 return m+tp;
```

```
}
int main()
{
 for(i=1;i<1000001;i++)
 {
  b[i]=i;
 }
 for(i=1;i<1000001;i++)
 {
  b[opot(i)]=0;
 }
 for(i=1;i<1000001;i++)
 {
  if(b[i]!=0)
    cout<<i<<endl;
 }
 return 0;
}
```

1108、FatMouse's Speed

```
#include <stdio.h>
#include <string.h>
int main()
{
```

```
  struct seta
  {
        int w;
        int s;
int hou;
int num;
  };

  seta a[11100];
  seta tp;
  int tt,tt2,s,w;
  int maxn;
  int n;
  int i,j;
  int b[11100];
  i=0;
  while (scanf("%d%d\n",&w,&s)!=EOF)
  {
       i++;
       a[i].w=w;
       a[i].s=s;
  a[i].num=i;
  }
```

```
n=i;
for (i=1;i<=n-1;i++)
for (j=i+1;j<=n;j++)
{



    if (a[i].w>a[j].w)
    {
        tp=a[i];
        a[i]=a[j];
        a[j]=tp;
    }
    if (a[i].w==a[j].w)
    {
                    if (a[i].s<a[j].s)
                    {
                                tp=a[i];
                                a[i]=a[j];
                                a[j]=tp;
                    }
    }
}
```

```
  memset(b,0,sizeof(b));
  a[n+1].s=-1;
  for (i=n;i>=1;i--)
  {
      maxn=0;
      for (j=i+1;j<=n+1;j++)
      {
            if ((a[j].s<a[i].s)&&(b[j]+1>=maxn)&&(a[i].w!=a[j].w))
            {
                  maxn=b[j]+1;
                  tt=j;
 }
      }
a[i].hou=tt;
      b[i]=maxn;
  }
  maxn=0;
  for (i=1;i<=n;i++)
  {
      if (b[i]>=maxn)
      {maxn=b[i];
 tt2=i;
}
```

```
    }
    printf("%d\n",maxn);
    do
  {
        printf("%d\n",a[tt2].num);
  tt2=a[tt2].hou;
  }
  while (tt2!=n+1);
  return 0;

}
```

1003、 Hike on a Graph

```
#include <iostream>
#include <queue>
#include <string.h>
using namespace std;
class pt
{
public:
 int i, k, j, c;
};
queue<pt> pq, tq, ttq;
```

```
int n, p1, p2, p3, b[50][50][50], ans;
char a[50][50];

void init()
{
 p1 --, p2 --, p3 --;
 memset(b, 0, sizeof(b));
 ans = -1;
 pq = tq;
 b[p1][p2][p3] = 1;
 pt tpt;
 tpt.i = p1, tpt.k = p2, tpt.j = p3, tpt.c = 0;
 pq.push(tpt);
}
void bfs()
{
 while(!pq.empty())
 {
   pt tpt, ttpt;
   int i, k, j;
   tpt = pq.front();
   if(tpt.i == tpt.k && tpt.k == tpt.j)
   {
```

```
 ans = tpt.c;
 break;
}
for(i = 0; i < n; i ++)
{
 if(a[tpt.j][tpt.k] == a[tpt.i][i])
 {
  k = tpt.k, j = tpt.j;
  if(!b[i][k][j])
  {
   b[i][k][j] = 1;
   ttpt.i = i, ttpt.j = j, ttpt.k = k, ttpt.c = tpt.c + 1;
   pq.push(ttpt);
  }
 }
}
for(k = 0; k < n; k ++)
{
 if(a[tpt.j][tpt.i] == a[tpt.k][k])
 {
  i = tpt.i, j = tpt.j;
  if(!b[i][k][j])
  {
```

```
        b[i][k][j] = 1;
        ttpt.i = i, ttpt.j = j, ttpt.k = k, ttpt.c = tpt.c + 1;
        pq.push(ttpt);
       }
      }
     }
     for(j = 0; j < n; j ++)
     {
      if(a[tpt.i][tpt.k] == a[tpt.j][j])
      {
       k = tpt.k, i = tpt.i;
       if(!b[i][k][j])
       {
        b[i][k][j] = 1;
        ttpt.i = i, ttpt.j = j, ttpt.k = k, ttpt.c = tpt.c + 1;
        pq.push(ttpt);
       }
      }
     }
     pq.pop();
   }
}
```

```
int main()
{
 while(cin >> n && n)
 {
  cin >> p1 >> p2 >> p3;
  int i, k;
  for(i = 0; i < n; i ++)
  {
   for(k = 0; k < n; k ++)
   {
    cin >> a[i][k];
   }
  }
  init();
  bfs();
  if(ans != -1)
   printf("%d\n", ans);
  else
   printf("impossible\n");
 }
 return 0;
}
```

1101、Gamblers

```
#include <iostream>

#include <algorithm>

using namespace std;



int c[1024];

bool opot(int c[],int h,int r,int t)

{

    int mid;

    if (h>r)

  return false;

    mid=(r-h)/2+h;

    if (c[mid]>t)

        return opot(c,h,mid-1,t);

    else if (c[mid]<t)

        return opot(c,mid+1,r,t);

    else

  return true;

}



int main()

{

    int n,i,j,k,t,win;
```

```
bool flag;
while (cin>>n && n!=0)
{
    for (i=0;i<n;i++)
        cin>>c[i];
    sort(c,c+n);
    flag=false;
    for (i=n-1;i>=0;i--)
        for(j=n-1;j>=0;j--)
        {
            if (c[j]+c[0]+c[1]>c[i]) continue;
            if (j!=i)
                for (k=j-1;k>=0;k--)
                {
                    if (k!=i)
                    {
                        t=c[i]-c[j]-c[k];
                        if (opot(c,0,k-1,t))
                        {
                            flag=true;
                            win=i;
                            goto output;
                        }
```

```
                    }
                }
            }
        output:
            if (flag) cout<<c[win]<<endl;
            else cout<<"no solution"<<endl;
    }
    return 0;
}
```

1092、Arbitrage

```
#include <iostream>
#include <fstream>
#include <cstring>
#include <string>
#include <map>
using namespace std;
double a[30][30];
map<string, int> sm, dm;
int n, c = 0, ac;
int main()
{
 //ifstream cin("in.txt");
 while(cin >> n && n)
```

```
{
 sm = dm;
 int i, k, t, j;
 for(i = 0; i < n; i ++)
 {
  string s;
  cin >> s;
  sm[s] = i;
  for(k = 0; k < n; k ++)
  {
   a[i][k] = 0;  //清零
  }
 }
 cin >> t;
 for(i = 0; i < t; i ++)
 {
  string s0, s1; double td;
  cin >> s0 >> td >> s1;
  a[sm[s0]][sm[s1]] = td;
 }
 //pt();
 for(j = 0; j < n; j ++)
 {
```

```
 for(i = 0; i < n; i ++)
 {
  for(k = 0; k < n; k ++)
  {
   if(a[i][j] * a[j][k] > a[i][k])
    a[i][k] = a[i][j] * a[j][k];     //Floyd
  }
 }
}
//pt();
ac = 0;
for(i = 0; i < n; i ++)
{
 if(a[i][i] > 1)
 {
  ac = 1;
  break;
 }
}
printf("Case %d: ", ++ c);
if(ac)
 printf("Yes\n");
else
```

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

1027、Human Gene Functions

```
#include <cstdio>
#include <string>
int a[5][5] = { {5, -1, -2, -1, -3}, {-1, 5, -2, -3, -4}, {-2, -3, 5, -2, -2},
{-1, -2, -2, 5, -1}, {-3, -4, -2, -1, 0} }, b[101][101], s0[101], s1[101];
int n, m, t, ans;
int mx(int a, int b, int c)
{
  if(a >= b && a >= c)
    return a;
  if(b >= a && b >= c)
    return b;
  if(c >= a && c >= b)
    return c;
  else
    return -99999999;
}
void read()
{
```

```
 char tc; int k;
 scanf("%d ", &n);
 for(k = 0; k < n; k ++)
 {
  scanf("%c", &tc);
  if(tc == 'A') s0[k] = 0;
  if(tc == 'C') s0[k] = 1;
  if(tc == 'G') s0[k] = 2;
  if(tc == 'T') s0[k] = 3;
 }
 scanf("%d ", &m);
 for(k = 0; k < m; k ++)
 {
  scanf("%c", &tc);
  if(tc == 'A') s1[k] = 0;
  if(tc == 'C') s1[k] = 1;
  if(tc == 'G') s1[k] = 2;
  if(tc == 'T') s1[k] = 3;
 }
}
void init()
{
 memset(b, 0, sizeof(b));
```

```
 int i, sum = 0;
 for(i = 1; i <= n; i ++)
 {
  sum += a[s0[i - 1]][4];
  b[i][0] = sum;
 }
 sum = 0;
 for(i = 1; i <= m; i ++)
 {
  sum += a[s1[i - 1]][4];
  b[0][i] = sum;
 }
 ans = -99999999;
}
void dp()
{
 int i, k;
 for(i = 1; i <= n; i ++)
 {
  for(k = 1; k <= m; k ++)
  {
   int ta = b[i - 1][k - 1] + a[s0[i - 1]][s1[k - 1]];
   int tb = b[i][k - 1] + a[4][s1[k - 1]];
```

```
    int tc = b[i - 1][k] + a[s0[i - 1]][4];
    b[i][k] = mx(ta, tb, tc);
   }
  }
  ans = b[n][m];
}
void ps()
{
  int i;
  for(i = 0; i < n; i ++)
  {
   printf("%d ", s0[i]);
  }
  printf("\n");
  for(i = 0; i < m; i ++)
  {
   printf("%d ", s1[i]);
  }
  printf("\n");
}
void pb()
{
  int i, k;
```

```
 for(i = 0; i <= n; i ++)
 {
  for(k = 0; k   <= m; k ++)
  {
    printf("%d ", b[i][k]);
  }
  printf("\n");
 }
 printf("\n");
}
int main()
{
 //freopen("in.txt", "r", stdin);
 scanf("%d", &t);
 int i;
 for(i = 0; i < t; i ++)
 {
  read();
  init();
  dp();
  //pb();
  printf("%d\n", ans);
 }
```

```
 return 0;
}
```

1082、 Stockbroker Grapevine

```
 #include <cstdio>
#include <string>
#define MX 1001
int a[100][100], n, m;
int main()
{
 //freopen("in.txt", "r", stdin);
 while(scanf("%d", &n) && n)
 {
  int i, k, j, ac = 1;
  for(i = 0; i < n; i ++)
  {
   scanf("%d", &m);
   for(k = 0; k < n; k ++)
   {
    a[i][k] = MX;
   }
   for(k = 0; k < m; k ++)
   {
```

```
  int ti, tn;
  scanf("%d %d", &ti, &tn);
  a[i][ti - 1] = tn;
 }
}
//pa();
for(j = 0; j < n; j ++)
{
 for(i = 0; i < n; i ++)
 {
  for(k = 0; k < n; k ++)
  {
   if(i == k || i == j || j == k) continue;
   if(a[i][j] + a[j][k] < a[i][k])
    a[i][k] = a[i][j] + a[j][k];
  }
 }
}
//pa();
int ans = MX, num;
for(i = 0; i < n; i ++)
{
 int mx = 0;
```

```
    for(k = 0; k < n; k ++)
    {
     if(i == k) continue;
     if(a[i][k] > mx)
      mx = a[i][k];
    }
    if(mx < ans)
    {
     ans = mx;
     num = i + 1;
    }
   }
   if(ans == MX)
    printf("disjoint\n");
   else
    printf("%d %d\n", num, ans);
  }
 return 0;
}
```

2109、FatMouse' Trade

```
#include <cstdio>
int n, m, a[1000], b[1000];
double r[1000];
```

```
void pt()
{
 for(int i = 0; i < m; i ++)
 {
  printf("%d %d %0.3f\n", a[i], b[i], r[i]);
 }
}
int main()
{
 //freopen("in.txt", "r", stdin);
 while(scanf("%d %d", &n, &m) && n != -1)
 {
  int i, k, j;
  for(i = 0; i < m; i ++)
  {
   scanf("%d %d", &a[i], &b[i]);
   if(b[i] == 0) r[i] = 99999999;
   else
    r[i] = double(a[i]) / double(b[i]);
  }
  //select sort
  for(i = 0; i < m - 1; i ++)
  {
```

```
 double mx = r[i]; j = i;
 for(k = i + 1; k < m; k ++)
 {
  if(r[k] > mx)
  {
   j = k;
   mx = r[k];
  }
 }
 int ai = a[i], bi = b[i]; double rd = r[i];
 a[i] = a[j], b[i] = b[j]; r[i] = r[j];
 a[j] = ai, b[j] = bi; r[j] = rd;
}
//pt();
double sum = 0; int si = n;
for(i = 0; i < m; i ++)
{
 if(si >= b[i])
 {
  si -= b[i];
  sum += double(a[i]);
 }
 else
```

```
    {
      sum += r[i] * double(si);
      si = 0;
    }
    if(si == 0)
      break;
  }
  printf("%0.3f\n", sum);
 }
 return 0;
}
```

1619、Present

```
#include <cstdio>
int m, n;
double a[101];
int main()
{
 //freopen("in.txt", "r", stdin);
 a[1] = 0, a[2] = 1;
 int i;
 for(i = 3; i < 101; i ++)
 {
  a[i] = (i - 1) * (a[i - 1] + a[i - 2]);
```

```
}
while(scanf("%d %d", &n, &m) != EOF)
{
 double pb = 1;
 if(n == m)
 {
  //pb *= a[n];
  for(i = 1; i <= n; i ++)
  {
   pb /= double(i);
  }
 }
 else
 {
  pb *= a[n - m];
  for(i = 1; i <= m; i ++)
  {
   pb /= double(i);
  }
  for(i = 1; i <= n - m; i ++)
  {
   pb /= double(i);
  }
```

```
  }
  printf("%0.8f\n", pb);
 }
 return 0;
}
```

1196、 Fast Food

```
#include <cstdio>
#include <string>
#include <cmath>
int d[201][201], s[201][201], a[201], n, m;
#define MX 99999999
int main()
{
 //freopen("in.txt", "r", stdin);
 int c = 0;
 while(scanf("%d %d", &n, &m) && n)
 {
  printf("Chain %d\n", ++ c);
  int i, k, j;
  for(i = 0; i < n; i ++)
  {
   scanf("%d", &a[i]);
  }
```

```
memset(s, 0, sizeof(s));
for(i = 0; i < n; i ++)
{
 for(k = i; k < n; k ++)
 {
  int mid = (i + k) / 2;
  for(j = i; j <= k; j ++)
  {
   s[i][k] += abs(a[j] - a[mid]);
  }
 }
}
for(i = 0; i < m; i ++)
{
 for(k = 0; k < n; k ++)
 {
  d[i][k] = MX;
 }
}
for(i = 0; i < n; i ++)
{
 d[0][i] = s[0][i];
}
```

```
  for(i = 1; i < m; i ++)
  {
   for(k = i; k < n; k ++)
   {
    if(k == i)
    {
     d[i][k] = 0;
     continue;
    }
    for(j = i - 1; j < k; j ++)
    {
     if(d[i - 1][j] + s[j + 1][k] < d[i][k])
      d[i][k] = d[i - 1][j] + s[j + 1][k];
    }
   }
  }
  printf("Total distance sum = %d\n", d[m - 1][n - 1]);
  printf("\n");
 }
 return 0;
}
```

1524、Supermarket

```
#include <cstdio>
```

```
#include <string>

double a[100];

int b[100], n, m, mx, tmx;

int main()

{

 int i, k;

 //freopen("in.txt", "r", stdin);

 while(scanf("%d %d", &n, &m) && n)

 {

  memset(a, 0, sizeof(a));

  for(i = 0; i < n; i ++)

  {

   scanf("%d", &b[i]);

  }

  mx = 0, tmx = mx;

  for(i = 0; i < m; i ++)

  {

   int ti; double td;

   scanf("%d %lf", &ti, &td);

   mx = tmx;

   for(k = mx; k >= 0; k --)

   {

    if(ti == b[k])
```

```
    {
     if(a[k] == 0 || a[k - 1] + td < a[k])
     {
      a[k] = td + a[k - 1];
      if(k == mx && mx < n - 1)
       tmx = mx + 1;
     }
    }
   }
  }
  if(a[n - 1] != 0)
   printf("%0.2f\n", a[n - 1]);
  else
   printf("Impossible\n");
 }
 return 0;
}
```

1366、Cash Machine

```
#include <cstdio>
#include <string>
int a[100001], b[100001], c, n, mx;
int ni[10], di[10];
int main()
```

```
{
 //freopen("in.txt", "r", stdin);
 while(scanf("%d", &c) != EOF)
 {
  int i, k, j;
  scanf("%d", &n);
  for(i = 0; i < n; i ++)
  {
   scanf("%d %d", &ni[i], &di[i]);
  }
  for(i = 0; i < n - 1; i ++)
  {
   int j = i, t0 = di[i], t1 = ni[i];
   for(k = i + 1; k < n; k ++)
   {
    if(t0 > di[k])
    {
     j = k; t0 = di[k];
    }
   }
   di[j] = di[i]; di[i] = t0; ni[i] = ni[j]; ni[j] = t1;
  }
  //pt();
```

```
mx = 0;
memset(a, 0, sizeof(a));
b[0] = 0;
int pi = 1, ti; // index for b[]....vector.
for(i = 0; i < n; i ++)
{
 ti = pi;
 for(j = 0; j < pi; j ++)
 {
  for(k = 0; k <= ni[i]; k ++)
  {
   int t = b[j] + di[i] * k;
   if(t <= c)
   {
    if(a[t]) continue;
    a[t] = 1;
    b[ti ++] = t;
    if(t == c)
     goto out;
   }
  }
 }
pi = ti;
```

```
  }
out:
  for(i = c; i >= 0; i --)
  {
   if(a[i])
   {
    mx = i;
    break;
   }
  }
  printf("%d\n", mx);
 }
 return 0;
}
```

1986、Bridging Signals

```
#include <cstdio>
#include <string>
int a[40000], c;
int main()
{
 int m, n, i, k;
 //freopen("in.txt", "r", stdin);
 scanf("%d", &m);
```

```
for(i = 0; i < m; i ++)
{
 memset(a, 0, sizeof(a));
 scanf("%d", &n);
 c = 0;
 for(k = 0; k < n; k ++)
 {
  int t;
  scanf("%d", &t);
  if(c == 0 || t > a[c - 1])
   a[c ++] = t;
  else
  {
   int l = 0, h = c - 1, mid = (l + h) / 2;
   while(l < h)
   {
    if(a[mid] < t) l = mid + 1;
    else if(a[mid] > t) h = mid;
    mid = (l + h) / 2;
   }
   a[mid] = t;
  }
  //pa();
```

```
  }
  printf("%d\n", c);
 }
 return 0;
}
```

1530、Find The Multiple

```
#include <cstdio>
int n, t, a[100], ac;
void dfs(int c, int s)
{
 if(!s && !ac)
 {
  ac = 1;
  for(int i = 0; i < c; i ++)
  {
   printf("%d", a[i]);
  }
  printf("\n");
 }
 else
 {
  if(c < 100 && !ac)
  {
```

```
    a[c] = 1;
    dfs(c + 1, (s * 10 + 1) % n);
    a[c] = 0;
    dfs(c + 1, (s * 10) % n);
  }
 }
}

int main()
{
 while(scanf("%d", &n) && n)
 {
  a[0] = 1, ac = 0;
  dfs(1, 1);
 }
 return 0;
}
```

2042、Divisibility

```
#include <cstdio>
#include <string>
int a[100], b[100], m, n, k;
int main()
{
```

```
int i, j, u;
//freopen("div.16", "r", stdin);
//freopen("in.txt", "r", stdin);
scanf("%d", &m);
for(i = 0; i < m; i ++)
{
 memset(a, 0, sizeof(a));
 memset(b, 0, sizeof(b));
 if(i) printf("\n");
 scanf("%d %d", &n, &k);

 int t;
 scanf("%d", &t);
 t %= k;
 if(t < 0) t += k;
 a[t] = 1;
 for(j = 1; j < n; j ++)
 {
  scanf("%d", &t);
  for(u = 0; u < k; u ++)
  {
   if(a[u])
   {
```

```
      int l = (u + t) % k;
      int r = (u - t) % k;
      if(l < 0) l += k;
      if(r < 0) r += k;
      b[l] = 1;
      b[r] = 1;
     }
    }
    //memset(a, 0, sizeof(a));
    memcpy(a, b, sizeof(b));
    memset(b, 0, sizeof(b));
   }
  if(a[0])
    printf("Divisible\n");
  else
    printf("Not divisible\n");
 }
 return 0;
}
```

2271、Chance to Encounter a Girl

```
#include <cstdio>
#include <string>
double p[101][100][100];
```

```
int pos[100][100], n;
int dir[4][2] = {{1, 0}, {-1, 0}, {0, 1}, {0, -1}};
void init()
{
 for(int i = 0; i < n; i ++)
 {
  for(int k = 0; k < n; k ++)
  {
   pos[i][k] = 4;
   if(i == 0 || i == n - 1)
    pos[i][k] --;
   if(k == 0 || k == n - 1)
    pos[i][k] --;
  }
 }
}
int main()
{
 //freopen("in.txt", "r", stdin);
 while(scanf("%d", &n) != EOF)
 {
  memset(p, 0, sizeof(p));
  init();
```

```
int t, i, k, u;
double p_p = 1.0000, s_p = 0.0000;
if((n - 3) % 4 == 0)
{
p[0][n / 2][n / 2] = p_p;
for(t = 1; t <= n; t ++)
{
 for(i = 0; i < n; i ++)
 {
  for(k = 0; k < n; k ++)
  {
   p_p = 0.0000;
   for(u = 0; u < 4; u ++)
   {
    int ii = i + dir[u][0];
    int kk = k + dir[u][1];
    if(kk >= 0 && kk < n && ii >= 0 && ii < n)
    {
     p_p += p[t - 1][ii][kk] / pos[ii][kk];
    }
   }
   p[t][i][k] = p_p;
  }
```

```
    }
    s_p += p[t][n / 2][t - 1];
    p[t][n / 2][t - 1] = 0;
   }
   }
   //pt();
   printf("%0.4f\n", s_p);
 }
 return 0;
}
```

1101、Gamblers

```
#include <stdio.h>
#include <stdlib.h>
int a[1000], n, wi, wj, wk, wu;
int comp(const void *a, const void *b)
{
    int aa = *(int*)a, bb = *(int*)b;
    return aa > bb;
}
int main()
{
 //freopen("in.txt", "r", stdin);
 while(scanf("%d", &n) && n)
```

```
{
 int i, k, j, u;
 for(i = 0; i < n; i ++)
  scanf("%d", &a[i]);
 qsort(a, n, sizeof(int), comp);
 wi = 536870912;
 if(n < 4)
  goto out;
 //o(n^3 * logn)
 for(i = n - 1; i >= 0; i --)
 {
  for(k = n - 1; k >= 0; k --)
  {
   if(i == k) continue;
   for(j = k - 1; j > 0; j --)
   {
    if(j == i) continue;
    u = a[i] - a[k] - a[j];
    if(u != a[i] && u != a[k] && u != a[j])
    {
     //if not equivalent then begin binary search
     int h = n - 1, l = 0, mid;
     while(l <= h)
```

```
      {
       mid = (h + l) / 2;
       if(u > a[mid]) l = mid + 1;
       else if(u < a[mid]) h = mid - 1;
       else if(u == a[mid])
       {
        wi = a[i], wk = a[k], wj = a[j], wu = u;
        goto out;
       }
      }
     }
    }
   }
out: if(wi == 536870912)
    printf("no solution\n");
   else
    //printf("%d %d %d %d\n", wi, wk, wj, wu);
    printf("%d\n", wi);
 }
 return 0;
}
```

2180、City Game

```
#include <cstdio>
#include <string>
/* state: 0.34s, 404kb */
int n, a, b;
int m[1000], r[1000], l[1000];
void pm()
{
 for(int i = 0; i < b; i ++)
 {
  printf("%d ", m[i]);
 }
 printf("\n");
}
int main()
{
 //freopen("in.txt", "r", stdin);
 int i, k, j;
 scanf("%d", &n);
 for(i = 0; i < n; i ++)
 {
  memset(m, 0, sizeof(m));
  scanf("%d %d ", &a, &b);
```

```
//pm();
int max = 0;
//dp(n^2.....)
for(k = 0; k < a; k ++)
{
 for(j = 0; j < b; j ++)
 {
  char tc; scanf("%c ", &tc);
  if(tc == 'R')
   m[j] = 0;
  else
   m[j] ++;
 }
 //pm();
 for(j = 0; j < b; j ++)
 {
  l[j] = j;
  while(l[j] > 0 && m[j] <= m[l[j] - 1])
   l[j] = l[l[j] - 1];
 }
 for(j = b - 1; j >= 0; j --)
 {
  r[j] = j;
```

```
    while(r[j] < b - 1 && m[j] <= m[r[j] + 1])
      r[j] = r[r[j] + 1];
    if(max < (r[j] - l[j] + 1) * m[j])
      max = (r[j] - l[j] + 1) * m[j];
   }
  }
  printf("%d\n", max * 3);
 }
 return 0;
}
```

1008、Gnome Tetravex

```
#include <stdio.h>
int g=0;          //Game index
int n=0;          //Puzzle size
int q=0;          //How many different types of squares
int square[25][4];    //Source squares
int count[25];       //Quantity of a certain type of squares
int table[25];       //Solution
int place(int pos)
{
    int i;
    if(pos==n*n)
       return 1;
```

```
    for(i=0; i<q; i++)
    {
      if(count[i]==0)
          continue;
      if(pos%n!=0)
          if(square[table[pos-1]][1]!=square[i][3])
            continue;
      if(pos/n!=0)
          if(square[table[pos-n]][2]!=square[i][0])
            continue;
      table[pos]=i;
      count[i]--;
      if(place(pos+1)==1)
          return 1;
      count[i]++;
    }
    return 0;
}
int main()
{
    int i, j;
    int t, r, b, l;    //Temporary variables for input (top, right, bottom, left)
    g=0;
```

```
q=0;
while(1)
{
  g++;
  scanf("%d", &n);
  if(n==0)
      break;
  q=0;
  for(i=0; i<n*n; i++)
  {
      scanf("%d %d %d %d", &t, &r, &b, &l);
      j=0;
      while(j<q)
      {
        if(square[j][0]==t && square[j][1]==r && square[j][2]==b && square[j][3]==l)
        {
            count[j]++;
            break;
        }
        j++;
      }
      if(j==q)
      {
```

```
            square[j][0]=t;
            square[j][1]=r;
            square[j][2]=b;
            square[j][3]=l;
            count[j]=1;
            q++;
          }
       }
       if(g>1)
          printf("\n");
       if(place(0)==1)
          printf("Game %d: Possible\n", g);
       else
          printf("Game %d: Impossible\n", g);
    }
    return 0;
}
```

1489、2^x mod n = 1

```
#include<stdio.h>

int modular(int a,long b,int n)
{
    long d,t ;
```

```
    d=1 ;
    t=a ;
    while(b>0)
    {
        if(b%2==1)
        d=d*t%n ;
        b=b/2 ;
        t=t*t%n ;
    }
    if(d==1)
    return 1 ;
    else return 0 ;
}

int main()
{
    long n ;
    long i ;
    while(scanf("%ld",&n)!=EOF)
    {
        if((n%2==0)||(n==1))
        {
            printf("2^? mod %ld = 1\n",n);
```

```
            continue ;
        }
        for(i=1;;i++)
        if(modular(2,i,n))
        {
            printf("2^%ld mod %ld = 1\n",i,n);
            break ;
        }

    }
    return 0 ;
}
```

1089、Lotto

```
#include <iostream>
using namespace std;

int l, s[13], d[13];

void print()
{
    int i, total;
    total=0;
    for(i=0; i<l; i++){
```

```
        if(s[i]){
            cout << d[i];
            if(total<5) cout << " ";
            total++;
        }
    }
    cout << endl;
}

void init()
{
    int i;
    for(i=0; i<l; i++){
        s[i]=0;
        d[i]=0;
    }
}

void build(int a, int b)
{
    int i;
    for(i=b; i<l; i++){
        if(s[i]==0){
```

```
            s[i]=1;
            if(a==6) print();
            else build(a+1, i+1);
            s[i]=0;
        }
    }
}

int main()
{
    int i;
    cin >> l;
    while(l){
        init();
        for(i=0; i<l; i++){
            cin >> d[i];
        }
        build(1, 0);
        cin >> l;
        if(l) cout << endl;
    }
    return 0;
}
```

1061、Web Navigation

```
#include <iostream.h>
#include <string.h>

int main()
{
    char data[120][100];
    int top;
    int now;
    char temp[100];
    int test;
    int k;
    cin >> test;
    for(k = 1; k <= test; ++k)
    {
        top = 0;
        strcpy(data[top],"http://www.acm.org/");
        now = top;
        while(1)
        {
            cin >> temp;
            if(temp[0] == 'Q')
                break;
```

```
else if(temp[0] == 'B')
{
    --now;
    if(now >= 0)
        cout << data[now] << endl;
    else
    {
        cout << "Ignored" << endl;
        now = 0;
    }
}
else if(temp[0] == 'F')
{
    ++now;
    if(now > top)
    {
        cout << "Ignored" << endl;
        now = top;
    }
    else
        cout << data[now] << endl;
}
else if(temp[0] == 'V')
```

```
        {
            top = now;
            ++top;
            cin >> data[top];
            cout << data[top] << endl;
            now = top;
        }
        else
            break;
    }
    if(k != test)
        cout << endl;
  }
  return 0;
}
```

1045、HangOver

```
#include <iostream>
using namespace std;

int main()
{
    int i;
    float x,result;
```

```
    scanf("%f",&x);
    while(x!=0)
    {
        for(i=2,result=0;;i++)
        {
            result=result+1.00/i;
            if(result>x)
                break;
        }
        printf("%d card(s)\n",i-1);
        scanf("%f",&x);
    }
}
```

1042、 W's Cipher

```
#include <iostream.h>
#include <string.h>

struct group
{
    char c;
    int p;
};
```

```
int main()
{
    int i,j,n,k1,k2,k3,ptr1,ptr2,ptr3;
    char a[82];
    group g1[82],g2[82],g3[82];
    while(cin>>k1>>k2>>k3)
    {
        if(!k1&&!k2&&!k3) break;
        cin.get();
        ptr1=ptr2=ptr3=0;
        cin.getline(a,82);
        n=strlen(a);
        for(i=0;i<n;i++)
            if(a[i]>='a'&&a[i]<='i')
                g1[ptr1].c=a[i],g1[ptr1++].p=i;
            else if(a[i]>='j'&&a[i]<='r')
                g2[ptr2].c=a[i],g2[ptr2++].p=i;
            else
                g3[ptr3].c=a[i],g3[ptr3++].p=i;

        for(i=0;i<ptr1;i++)
        {
            j=i+k1;
```

```
            if(j>=ptr1) j%=ptr1;
            a[g1[j].p]=g1[i].c;
        }
        for(i=0;i<ptr2;i++)
        {
            j=i+k2;
            if(j>=ptr2) j%=ptr2;
            a[g2[j].p]=g2[i].c;
        }
        for(i=0;i<ptr3;i++)
        {
            j=i+k3;
            if(j>=ptr3) j%=ptr3;
            a[g3[j].p]=g3[i].c;
        }
        for(i=0;i<n;i++) cout<<a[i];
        cout<<endl;
    }
    return 0;
}
```

1029、Moving Tables

```
#include <iostream>
using namespace std;
```

```
int P[200];
int N;

int main()
{
   int t;
   cin>>t;
   int i,j,s,d;
   for(i=0;i<t;i++)
      {

         cin>>N;
         for(j=0;j<200;j++)
            P[j]=0;
         for(j=0;j<N;j++)
            {
               cin>>s>>d;
               s=(s-1)/2;
               d=(d-1)/2;
               if(s>d)
                  {
                     int t=s;
```

```
                    s=d;
                    d=t;
                }
            for(int k=s;k<=d;k++)
                P[k]++;
        }
        int mm=-2;
        for(j=0;j<200;j++)
            if(P[j]>mm)
                mm=P[j];
        cout<<mm*10<<endl;
    }
  return 0;
}
```

1004、Anagrams by Stack

```
#include <iostream>
#include <stack>
#include <string>
using namespace std;

char * pSource, *pDest;
char Q[1000];
int QLen, m;
```

```
void TryIt(stack<char>&s)
{
    char ch;
    if(*pDest==0) // 找到一个解
    {
        for(m=0;m<QLen;++m) cout<<Q[m] << " ";
        cout << endl;
        return;
    }
    // 先尝试输入一个字母
    if(*pSource!=0)
    {
        s.push(*pSource++);
        Q[QLen++] = 'i';
        TryIt(s);
        s.pop();
        QLen--;
        pSource--;
    }
    // 再尝试输出一个字母
    if(!s.empty())
    {
```

```
        ch = s.top();
        if(*pDest==ch)
        {
            s.pop();
            Q[QLen++] = 'o';
            pDest++;
            TryIt(s);
            pDest--;
            QLen--;
            s.push(ch);
        }
    }
}
int main()
{
    string strSource, strDest;
    stack<char> sTemp;
    while(cin>>strSource>>strDest)
    {
        QLen = 0;
        pSource = (char*)strSource.c_str();
        pDest = (char*)strDest.c_str();
        cout<<"["<<endl;
```

```
        TryIt(sTemp);
        cout<<"]"<<endl;
    }
    return 0;
}
```

1003、Crashing Balloon

```
#include <iostream>
#include <cmath>
using namespace std;

const int MAX = 100;
bool UsedBalloon[MAX+1];

bool CanFactor(int i, int b)
{
    if (b == 1)
    {
        return true;
    }
    else
    {
        while (i <= MAX)
        { // find a possible factor of b
```

```
            if ( (!UsedBalloon[i]) && ((b % i) == 0) )
            {
                break;
            }
            else
            {
                i++;
            }
        }
        if (i > MAX) return false;

            // try the factor i
        if (CanFactor(i+1, b / i))
        {
            return true;
        }
        else
        {
            // i can not be a factor
            return CanFactor(i + 1, b);
        }
    }
}
```

```
bool IsPossible(int i, int a, int b)
{
    if (i > MAX) return false;
    if (a == 1) return (CanFactor(1, b)); // try to factor b

    while (i <= MAX)
    { // find a possible factor of a
        if ((a % i) == 0)
        {
            break;
        }
        else
        {
            i++;
        }
    }
    if (i > MAX) return false;
    UsedBalloon[i] = true;
    if (IsPossible(i + 1, a / i, b)) return true; // try a factor i of a
    UsedBalloon[i] = false;
    return (IsPossible(i + 1, a, b)); // try not use i as factor of a
}
```

```
int GetWinner(int a, int b)
{
    for (int i = 0; i < MAX + 1; i++)
    {
        UsedBalloon[i] = false;
    }

    if (a < b)
    { // make sure b is the chanllenge
        int tmp = a;
    a = b;
    b = tmp;
    }
    if ((CanFactor(1, b) == false) && (b > 100))
    { // chanllenge is lying.
        return a;
    }
    else if (IsPossible(1, a, b))
    { // is it possible?
        return a; // chanllenge fail.
    }
    else
```

```
    {
        return b; // it's impossible, chanllenge win.
    }
}

int main()
{
    int a, b;
    while (cin >> a >> b)
    {
        cout << GetWinner(a, b) << endl;
    }
    return 0;
}
```

1100、 Mondriaan's Dream

```
#include <stdio.h>
#include <string>
double ans , F[12][2049];
int   height , wide;
bool record[13];
bool   DFS_jieya (int start)
{
            if (start == 0) return true;
```

```
        int i;
        for (i=wide-1;i>=0;i--)
        {
            if (record[i] == false && start - (1<<i) >= 0)
            {
                record[i] = true;
                if (DFS_jieya (start - (1<<i))) return true;
                record[i] = false;
            }
        }
    return false;

}
void Construct (int s , int j , int k , int begin) //F[s][j] --->> F[s+1][k];
{
        int i;
        for (i=begin;i<=wide-1;i++)
        {
            if (record[i] == false)
            {
                if (i+1 <= wide-1 && record[i+1] == false)
                {
                    Construct (s , j , k , i+2);
```

```
            }


              Construct (s , j , k+(1<<i) , i+1);

               return ;

            }

        }


      // printf ("F[%d][%d] = %d ---- >>> F[%d][%d]\n" , s , j , F[s][j] , s+1 , k);

      F[s+1][k] += F[s][j];


 return ;

}


void Construct_LastLine (int j , int k , int begin) // F[height][j] --->> ans += F[height][j];

{

        int i;

       for (i=begin;i<=wide-1;i++)

       {

            if (record[i] == false)

            {

                if (i+1 <= wide-1 && record[i+1] == false)

                {

                    Construct_LastLine (j , k , i+2);
```

```
                }
                return ;
            }
        }
    // printf ("F[%d][%d] = %d\n" , height , j , F[height][j]);
     ans += F[height][j];
 return ;
}
void Solve ()
{
    int i , j , k;
    memset (F , 0 , sizeof(F));
    memset (record , false , sizeof(record));
     F[1][0] = 1;
    Construct (1 , 0 , 0 , 0);
    for (i=2;i<=height-1;i++)
    {
        for (j=0;j<=1<<wide;j++)
        {
            if (F[i][j] > 0)
            {
                memset (record , false , sizeof(record));
                if (DFS_jieya (j) == false) { printf ("ERROR!  ::  j = %d\n" , j); return ; }
```

```
                    Construct (i , j , 0 , 0);
                }
            }
        }
        for (j=0;j<=1<<wide;j++)
        {
            if (F[height][j] > 0)
            {
                memset (record , false , sizeof(record));
                DFS_jieya (j);
                Construct_LastLine (j , 0 , 0);
            }
        }
    return ;
}
int main ()
{
        while (scanf ("\n%d %d" , &height , &wide) != EOF)
        {
            if (height == 0 && wide == 0) break ;
            ans = 0;
            Solve ();
            printf ("%.0lf\n" , ans);
```

```
        }
 return 0;
}
```

1027、Human Gene Functions

```
#include <cstdio>
#include <string>
#include <iostream>
using namespace std;
#define MAX(a,b,c) (a>b?a:b)>c?(a>b?a:b):c
int ctoi(char a)
{
    int b;
    if(a=='A')          b = 0;
    if(a=='C')          b = 1;
    if(a=='G')          b = 2;
    if(a=='T')          b = 3;
    if(a=='-')          b = 4;
    return b;
}

int main()
{
    int t,j,k,m,n;
```

```
int f1,f2,f3;
int f[101][101];
int arr[5][5]=
{
	{ 5,-1,-2,-1,-3},
	{ -1,5,-3,-2,-4},
	{ -2,-3,5,-2,-2},
	{ -1,-2,-2,5,-1},
	{ -3,-4,-2,-1,0}
};
string a,b;
cin>>t;
while(t--)
{
	j = k = 0;
	memset(f,0,sizeof(f));
	cin>>m>>a;
	cin>>n>>b;
	for(j=0;j<=m;j++)
	{
		for(k=0;k<=n;k++)
		{
			if(j == 0 && k == 0)
```

```
            {
                f[j][k] = 0;
            }
            else if(j==0)
            {
                f[j][k] = f[j][k-1] + arr[ctoi('-')][ctoi(b[k-1])];
            }
            else if(k==0)
            {
                f[j][k] = f[j-1][k] + arr[ctoi(a[j-1])][ctoi('-')];
            }
            else
            {
                f1 = f[j-1][k]     + arr[ctoi(a[j-1])][ctoi( '-')];
                f2 = f[j][k-1]     + arr[ctoi( '-')][ctoi(b[k-1])];
                f3 = f[j-1][k-1] + arr[ctoi(a[j-1])][ctoi(b[k-1])];

                f[j][k] = MAX(f1,f2,f3);

            }
        }
}
cout<<f[m][n]<<endl;
```

```
    }

    return 0;

}
```

1097、Code the Tree

```
#include <cstdio>

#include <memory.h>

#include <cctype>

#define MAXN 100



int degree[MAXN];

bool adj[MAXN][MAXN];

int n;

int ni;

char line[1000];



void make(int root,int &k)

{

    if (root>n) n=root;

    while (line[k]=='(')

    {

        int now;

        k++;

        sscanf(line+k,"%d",&now);
```

```
            while (isdigit(line[k])) k++;
            degree[root]++;
            degree[now]++;
            adj[root][now]=adj[now][root]=true;
            make(now,k);
            k++;
        }
        return;
}

int main()
{
    char ch;
    int len;
    int i,j,k;
    while (scanf("%c",&ch)!=EOF)
    {
            len=0;
            k=0;
            memset(line,0,sizeof(line));
            memset(degree,0,sizeof(degree));
            memset(adj,false,sizeof(adj));
            while (ch!=10)
```

```
{
        while (ch==' ') scanf("%c",&ch);
        if (ch!=10)
        {
            line[len]=ch;
            len++;
            scanf("%c",&ch);
        }
}
n=0;
make(0,k);
for (i=1;i<=n;i++)
if (adj[0][i])
{
    adj[0][i]=adj[i][0]=false;
    degree[i]--;
    break;
}



ni=0;
while (++ni<n)
{
```

```
                for (i=1;i<=n;i++) if (degree[i]==1) break;
                for (j=1;j<=n;j++) if (adj[i][j])
                {
                    adj[i][j]=adj[j][i]=false;
                    degree[j]--;
                    printf("%d",j);
                    if (ni<n-1) putchar(' ');
                    break;
                }
                degree[i]--;
            }
            putchar('\n');
        }
        return 0;
}
```

1163、The Staircases

```
#include    <iostream>
    #include    <string>
    using    namespace    std;

    const    int    maxn=500;
    double    f[maxn+1][maxn+1];
```

```
int	dp()
{
f[1][1]=1;
f[2][2]=1;
int	i,j,k;
for(i=3;i<=maxn;i++)
{
f[i][i]=1;
for(j=0;j<=i/2;j++)
for(k=j+1;j+k<=i;k++)
f[i][k]+=f[i-k][j];
}
return	0;
}
int	main()
{
int	n;
dp();
cin	>>	n;
while	(n)
{
int	i;
double	s=0;
```

```
for(i=1;i<n;i++)
s+=f[n][i];
cout.setf(ios::fixed);
cout.precision (0);
cout << s << endl;
cin >> n;
}
return 0;
}
```

1170、String Matching

```
#include <iostream.h>
#include <string.h>
int main()
{
int i,j;
char a[100],b[100];
int sum,sum1;
int s2,w1,w2,w;
while(cin>>a)
{
  if(strcmp(a,"-1")==0)  return 0;  cin>>b;
 sum=0;
 for(j=0;j<strlen(a)+strlen(b);j++)
```

```
{
  sum1=0;
  i=j;
  if(j>strlen(a)-1) i=strlen(a)-1;
  while(i>=0)
  {
        if(a[i]==b[strlen(b)-j+i])sum1++;
        i--;
   }
   if(sum<sum1)    sum=sum1;
}
  cout<<"appx("<<a<<","<<b<<") = ";
  w1=strlen(a);
  w2=strlen(b);
  w=w1+w2;
  s2=2*sum;
  if(s2==0) cout<<"0\n";
  else if(s2==w) cout<<"1\n";
     else {
          for(i=2;i<=strlen(a);i++)
           {   while(w%i==0 && s2%i==0)
                  {              s2=s2/i;        w=w/i;    }}
    cout<<s2<<"/"<<w<<endl;              }}}
```

1218、Ratio

```
#include<iostream.h>
   #include<math.h>

   int     gcb(int     a,int     b)
   {
        if(a%b==0)return     b;
        else     return     gcb(b,a%b);
   }

   int     main()
   {
        long     a,b,k=0;
        double     a1,b1,c,t,x;
        for(;cin>>a>>b;)
        {
           x=a/(double)b;
           b=1;
           c=1e13;
           if(k==1)cout<<endl;
           k=1;
           for(;;b+=1)
           {
```

```
                a=1;
                t=1e13;
                a1=a;b1=b;
                while(fabs(a1/b1-x)<=t)
                {
                    t=fabs(a1/b1-x);
                    ++a;a1+=1;
                }
                if(fabs((a1-1)/b1-x)<c&&gcb(a-1,b1)==1){
                cout<<a-1<<'/'<<b<<endl;
                c=fabs((a1-1)/b1-x);
                }
                if(fabs((a1-1)/b1-x)<1e-13)
                break;
            }
        }
        return  0;
    }
```

1232、Adventure of Super Mario

```
#include  <iostream.h>
  #include  <string.h>
```

```
#ifdef	DEBUG
#include	<fstream.h>
#include	<iomanip.h>
#endif

using	namespace	std;

#ifdef	DEBUG
	ifstream	fin("1232.in");
	ofstream	fdebug("debug.txt");
	istream&	in	=	fin;
	ostream&	out	=	cout;
#else
	istream&	in	=	cin;
	ostream&	out	=	cout;
#endif

const	int	MAX_NODE	=	100;
const	int	MAX_K	=	10;
const	int	MAX_L	=	500;
const	int	HEAP_SIZE	=	MAX_NODE	*	MAX_K	+	100;

typedef	int	Graph[MAX_NODE][MAX_NODE];
```

```
class Heap
{
public:
    int data[HEAP_SIZE];
    int size;
    int index[HEAP_SIZE];
    int cost[HEAP_SIZE];

    void shift_up(int i)
    {
        int j;
        while (i > 0) {
            j = (i - 1) / 2;
            if ( cost[ data[i] ] < cost[ data[j] ]) {
                swap(index[ data[i] ], index[ data[j] ]);
                swap(data[i], data[j]);
                i = j;
            } else {
                break;
            }
        }
    }
```

```
void shift_down(int i)
{
    int j, k;

    while (2 * i + 1 < size) {
        j = 2 * i + 1;
        k = j + 1;

        if ( (k < size) && (cost[ data[k] ] < cost[ data[j] ]) &&
                (cost[ data[k] ] < cost[ data[i] ]) )
        {
            swap(index[ data[k] ], index[ data[i] ]);
            swap(data[k], data[i]);
            i = k;
        } else if (cost[ data[j] ] < cost[ data[i] ]) {
            swap(index[ data[j] ], index[ data[i] ]);
            swap(data[j], data[i]);
            i = j;
        } else {
            break;
        }
    }
```

```
}

void init() {
    size = 0;
    memset(index, -1, sizeof(index));
    memset(cost, -1, sizeof(cost));
}

bool empty() {
    return (size == 0);
}

int pop()
{
    int res = data[0];
    data[0] = data[size-1];
    index[ data[0] ] = 0;
    size--;
    shift_down(0);
    return res;
}
```

```
    int top() {
        return data[0];
    }

    void push(int x, int c)
    {
        if (index[x] == -1) {
            cost[x] = c;
            data[size] = x;
            index[x] = size;
            size++;
            shift_up(index[x]);
        } else if (c < cost[x]) {
            cost[x] = c;
            shift_up(index[x]);
            shift_down(index[x]);
        }
    }
};

////////////////////////////////////////

struct State {
```

```
        int     pos,    times,    id;
};

Graph    G,    V;
int    A,    B,    K,    L,    n,    m;
Heap    heap;

int    hash(const    State&    s)
{
        return    (s.times    *    MAX_NODE    +    s.pos);
}

State    dehash(int    x)
{
        State    s;
        s.id    =    x;
        s.pos    =    x    %    MAX_NODE;
        x    /=    MAX_NODE;
        s.times    =    x;
        return    s;
}

void    Folyd()
```

```
{
    int i, j, k;

    memset(V, -1, sizeof(V));

    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            V[i][j] = G[i][j];

        }
    }

    for (k = 0; k < A; k++) {
        for (i = 0; i < n; i++) {
            for (j = 0; j < n; j++) {
                if ((V[i][k] == -1) || (V[k][j] == -1)) continue;
                if ((V[i][j] == -1) || (V[i][k] + V[k][j] < V[i][j])) {
                    V[i][j] = V[i][k] + V[k][j];
                }
            }
        }
    }
}
```

```
void Solve()
{
    State s, ns;

    Folyd();

    heap.init();
    s.pos = n - 1;
    s.times = K;
    heap.push(hash(s), 0);

    while (! heap.empty()) {
        s = dehash(heap.pop());
        if (s.pos == 0) {
            out << heap.cost[s.id] << endl;
            return;
        }

        // not use super run
        for (int i = 0; i < n; i++) {
            if (G[s.pos][i] == -1) continue;
            ns.pos = i;
```

```
                ns.times = s.times;
                heap.push( hash(ns), heap.cost[s.id] + G[s.pos][i] );
            }

            // use super run
            if (s.times > 0) {
                for (int i = 0; i < n; i++) {
                    if (V[s.pos][i] == -1) continue;
                    if (V[s.pos][i] > L) continue;
                    ns.pos = i;
                    ns.times = s.times - 1;
                    heap.push( hash(ns), heap.cost[s.id] );
                }
            }

        }
        out << -1 << endl;
}

int main()
{
        int T, x, y;
```

```
    in >> T;
    while (T--) {
        in >> A >> B >> m >> L >> K;
        n = A + B;
        memset(G, -1, sizeof(G));
        for (int i = 0; i < m; i++) {
            in >> x >> y;
            in >> G[x-1][y-1];
            G[y-1][x-1] = G[x-1][y-1];
        }
        Solve();
    }
    return 0;
}
```

1858、Soundex

```
#include <iostream.h>
#include <string.h>
int fun(char a)
{
    if(a=='B' || a=='F' || a=='P' || a=='V')
        return 1;
    if(a=='C' || a=='G' || a=='J' || a=='K' || a=='Q'
        || a=='S' || a=='X' || a=='Z')
```

```
        return 2;
    if(a=='D' || a=='T')
        return 3;
    if(a=='L')
        return 4;
    if(a=='M' || a=='N')
        return 5;
    if(a=='R')
        return 6;
    return 0;
}
int main()
{
    char s[20];
    int i,bef;
    while(cin>>s)
    {
        i=0;
        bef=10;
        while(s[i]!='\0')
        {
            if(fun(s[i])!=0)
            {
```

```
                if(fun(s[i])!=bef)
                cout<<fun(s[i]);
            }
            bef=fun(s[i]);
            i++;
        }
        cout<<endl;
    }
}
```

1884、WERTYU

```
#include<stdio.h>
#include<string.h>
char a[4][13]={
{'`','1','2','3','4','5','6','7','8','9','0','-','='},
{'Q','W','E','R','T','Y','U','I','O','P','[',']','\\'},
{'A','S','D','F','G','H','J','K','L',';','\'','0','0'},
{'Z','X','C','V','B','N','M',',','.','/','0','0','0'}
};
char fun(char b)
{
int i,j;
for (i=0;i<4;i++)
for    (j=0;j<13;j++)
```

```
if (b==a[i][j])
return a[i][j-1];
}

int main()
{
int i,j;
char s[100];
/*for (i=0;i<4;i++)
{
for (j=0;j<13;j++)
printf("%c ",a[i][j]) ;
printf("\n");
}
getch();
*/
while(gets(s)!=NULL)
{
for (i=0;i<strlen(s);i++)
if (s[i]!=' ')
s[i]=fun(s[i]);
printf("%s\n",s);
}
```

```
}
```

1969、Hard to Believe, but True!

```
#include <stdio.h>
#include <string.h>

int main()
{
char a[8],b[8],c[8],d[8];
char op[25];
int pos,pos2;
int i,flag,k;
while(1)
{
scanf("%s",&op);
        pos=pos2=0;
        while(op[pos2]!='+')
        {
        a[pos]=op[pos2];
        pos++;
        pos2++;
        }
        a[pos]='\0';
        pos2++;
```

```
pos=0;
while(op[pos2]!='=')
{
b[pos]=op[pos2];
pos++;
pos2++;
}
b[pos]='\0';
pos2++;
pos=0;
while(op[pos2]!='\0')
{
c[pos]=op[pos2];
pos++;
pos2++;
}
c[pos]='\0';
if (strcmp(a,"0")==0 && strcmp(b,"0")==0 && strcmp(c,"0")==0)
{
printf("True\n");
return 0;
}
for(i=0;i<7;i++)
```

```
d[i]='0';
d[7]='\0';
for(i=0;i<strlen(a);i++)
d[6-i]=a[i];
strcpy(a,d);
for(i=0;i<7;i++)
d[i]='0';
d[7]='\0';
for(i=0;i<strlen(b);i++)
d[6-i]=b[i];
strcpy(b,d);
for(i=0;i<7;i++)
d[i]='0';
d[7]='\0';
for(i=0;i<strlen(c);i++)
d[6-i]=c[i];
strcpy(c,d);
flag=0;
for(i=6;i>=0;i--)
{
k=a[i]-'0'+b[i]-'0'+flag;
if (k>=10)
        {
```

```
            flag=1;
            k=k-10;
            }
            else
            flag=0;
        d[i]=k+'0';
        }
        d[7]='\0';
        if (strcmp(c,d)==0)
        printf("True\n");
        else
        printf("False\n");
}

return 0;
}
```

1981、 Drink, on Ice

```
#include <stdio.h>
#include <math.h>
int main()
{
float Cw=4.19;
float Ci=2.09;
```

```
int Em=335;
float a,b,c,d;
float t,Mw,Mi;
while(1)
{
scanf("%f%f%f%f",&a,&b,&c,&d);
d=-d;
if (a<0.001 && b<0.001 && c<0.001 && d<0.001) return 0;
if (a*c*Cw-b*d*Ci>0)
{
    if (a*c*Cw-b*d*Ci-b*Em>0)
    {
    Mw=a+b;
    Mi=0;
    t=(a*c*Cw-b*d*Ci-b*Em)/Cw/(a+b);
    }
    else
    {
    Mw=a+(a*c*Cw-b*d*Ci)/Em;
    Mi=b-(a*c*Cw-b*d*Ci)/Em;
    t=0;
    }
}
```

```
else
{
    if (b*d*Ci-a*c*Cw-a*Em>0)
    {
    Mw=0;
    Mi=a+b;
    t=-(b*d*Ci-a*c*Cw-a*Em)/Ci/(a+b);
    }
    else
    {
    Mw=a-(b*d*Ci-a*c*Cw)/Em;
    Mi=b+(b*d*Ci-a*c*Cw)/Em;
    t=0;
    }
}
printf("%0.1f g of ice and %0.1f g of water at %0.1f C\n",Mi,Mw,t);
}
}
```

4045、Divisor Summation

```
#include<iostream.h>
#include<string.h>
long a[500001];
int main()
```

```
{   int i,j,k,N,n;
    memset(a,0,sizeof(a));
    for(i=1;i<=250000;i++)
      {
          k=2;
          while(k*i<=500000)
            {
                a[k*i]+=i;
                k++;
            }
      }
      cin>>N;
      for(i=1;i<=N;i++)
        {
            cin>>n;
            cout<<a[n]<<endl;
        }
     return 0;
}
```

1268、 Is It A Tree

```
#include <stdio.h>
#include <string.h>
int p[1001], x, y, b[1001], ac, c[1001], k = 1;
```

```
int find_set(int i)
{
  if(p[i] != i)
  {
    p[i] = find_set(p[i]);
  }
  return p[i];
}

void proc()
{
  int i, t = 0;
  for(i = 0; i < 1001; i ++)
  {
    if(b[i])
    {
      c[find_set(i)] ++;
    }
  }

  for(i = 0; i < 1001; i ++)
  {
    if(c[i] > 1)
```

```
    t ++;
  }
  if(t > 1) ac = 0;
}

void init()
{
  int i;
  for(i = 0; i <= 1000; i ++)
    p[i] = i;
  ac = 1;
  memset(b, 0, sizeof(b));
  memset(c, 0, sizeof(c));
}
void pt()
{
  printf("Case %d is ", k ++);
  if(ac)
    printf("a tree.\n");
  else
    printf("not a tree.\n");
}
int main()
```

```
{
  //freopen("in.txt", "r", stdin);
  init();
  while(scanf("%d %d", &x, &y))
  {
    if(x == -1)
      break;
    if(x == 0)
    {
      proc();
      pt();
      init();
      continue;
    }
    b[x] = 1, b[y] = 1;
    if(p[y] != y || find_set(x) == y) ac = 0;
    p[y] = find_set(x);
  }
  return 0;
}
```

1110、Dick and Jane

```
#include <iostream>
using namespace std;
```

```
main()
{
    const int d=12;
    int s,p,y,j;
    float sa,pa,ya;
  while(   scanf("%d%d%d%d",&s,&p,&y,&j)!=EOF){
      for(ya=0.0;ya<=50;ya+=0.1) {
          for(pa=ya;pa<=ya+p+3;pa+=0.1) {
              for(sa=pa;sa<=ya+y+3;sa+=0.1)
              {
                  if ( (int(sa-pa)==s) && (int(sa-ya)==y) && (int(pa-ya)==p) && (int(sa)+int(pa)+int(ya)==d+j) )
                      goto finish;
              }
          }
      }
finish:
      printf("%.0f %.0f %.0f\n",sa,pa,ya);
  }
}
```

1986、Bridging Signals

```
#include <stdio.h>
int res[40000];
int binSearch(int left, int right, int num)
```

```
{
        while(left <= right)
        {
                int mid = (left + right) /2;
                if(res[mid] == num)
                        return mid;
                else if(res[mid] < num)
                        left = mid + 1;
                else
                        right = mid - 1;
        }
        return right;       // return the right pos if cant find the num
}
int main()
{
        int t, n, num;
        scanf("%d", &t);
        while(t--)
        {
                scanf("%d %d", &n, &num);
                res[0] = num;
                int tot = 1;
                for(int i=1;i<n;++i)
```

```
                {
                        scanf("%d", &num);
                int pos = binSearch(0, tot-1, num);
                        if(pos == tot - 1)
                          res[tot++] = num;
                        else r
                        res[pos+1] = num;
                }
                printf("%d\n", tot);
        }
        return 0;
}
```

1070、 Bode Plot

```
#include<stdio.h>
#include<math.h>
float Vs,R,C,w;
double vol()
{
    double x=Vs*R*C*w;
    return x/sqrt(1+x*x);
}
int main()
{
```

```
    int cases,mc;
    scanf("%f%f%f%d",&Vs,&R,&C,&cases);
    for(mc=1;mc<=cases;mc++)
    {
        scanf("%f",&w);
        printf("%.3f\n",vol());
    }
    return 0;
}
```

1421、Dolphin Pool

```
#include<iostream>
#include<iomanip>
#include<string>
#include<algorithm>
#include<math.h>
#include<assert.h>
using namespace std;
#define NEG 1e-7
struct intersect{
    int other;
    char sign;
    double x, y, x0, y0;
}p[20][40];
```

```
struct circle{
	int x,y,r;
}cir[20];
int n,nsec[20];
bool v[20][40];
bool operator<(const intersect& x, const intersect& y)
{
	return atan2(x.y-x.y0,x.x-x.x0)<atan2(y.y-y.y0,y.x-y.x0)-NEG;
}
int Partition(intersect b[],int l,int r,int a)
{
	intersect x=b[l],t;
	int i=l-1,j=r+1;
	while(1)
	{
		do{
			--j;
		}while(x<b[j]);
		do{
			++i;
		}while(b[i]<x);
		if(i<j){
p[b[i].other/100][b[i].other%100].other=a*100+j;
```

```
p[b[j].other/100][b[j].other%100].other=a*100+i;
                    t=b[i];b[i]=b[j];b[j]=t;
            }
            else return j;
        }
        return l;
}
void quicksort(intersect b[],int l,int r,int a)
{
        int n;
        if(l<r)
        {
            n=Partition(b,l,r,a);
            quicksort(b,l,n,a);
            quicksort(b,n+1,r,a);
        }
        return;
}

void GetPoint(int a,int b,intersect& p1,intersect& p2)
{
        int A,B,C,D,E,F;
        double A1,B1,C1;
```

```
        A=-2*cir[a].x;B=-2*cir[a].y;
        C=cir[a].x*cir[a].x+cir[a].y*cir[a].y-cir[a].r*cir[a].r;
        D=-2*cir[b].x;E=-2*cir[b].y;
        F=cir[b].x*cir[b].x+cir[b].y*cir[b].y-cir[b].r*cir[b].r;
        if(A==D)                                        {
                p1.y=p2.y=(F-C)/(double)(B-E);
                p1.x=(-A+sqrt((double)A*A-4*(p1.y*p1.y+B*p1.y+C)))/2.0;
p2.x=(-A-sqrt((double)A*A-4*(p2.y*p2.y+B*p2.y+C)))/2.0;
        }
        else if(B==E)
        {
                p1.x=p2.x=(F-C)/(double)(A-D);
p1.y=(-B+sqrt((double)B*B-4*(p1.x*p1.x+A*p1.x+C)))/2.0;
p2.y=(-B-sqrt((double)B*B-4*(p2.x*p2.x+A*p2.x+C)))/2.0;
        }
        else
        {
A1=(double)(E-B)*(E-B)+(double)(A-D)*(A-D);
                B1=2.0*(E-B)*(F-C)+((double)D)*(A-D)*(E-B)+(A-D)*(A-D)*(double)E;
C1=(double)(F-C)*(F-C)+((double)D)*(A-D)*(F-C)+((double)F)*(A-D)*(A-D);
                p1.y=(-B1+sqrt(B1*B1-4*A1*C1))/(2*A1);
                p2.y=(-B1-sqrt(B1*B1-4*A1*C1))/(2*A1);
                p1.x=(p1.y*(E-B)+(F-C))/(A-D);
```

```
            p2.x=(p2.y*(E-B)+(F-C))/(A-D);
        }
        return;
}
bool Cover(int k,intersect& p0)
{
        if((p0.x-cir[k].x)*(p0.x-cir[k].x)+(p0.y-cir[k].y)*(p0.y-cir[k].y)>cir[k].r*cir[k].r+NEG)
                return false;
        else
                return true;
}
int Visit(int i,int j)
{
        double lx,ly,s;
        s=0;lx=p[i][j].x;ly=p[i][j].y;
        while(!v[i][j])
        {
                v[i][j]=true;
                i=p[i][j].other;
                j=i%100;i/=100;
                v[i][j]=true;
                if(j<nsec[i]-1)j++;
                else j=0;
```

```
            s+=(lx*p[i][j].y-ly*p[i][j].x)/2;
            lx=p[i][j].x;ly=p[i][j].y;
        }
        if(s>-NEG)return 0;
        else return 1;
}
int main()
{
        int t,index,i,j,k;
        intersect p1,p2;
        cin>>t;
        for(index=1;index<=t;index++)
        {
            cin>>n;
            for(i=0;i<n;i++)
                cin>>cir[i].x>>cir[i].y>>cir[i].r;
            fill(nsec,nsec+n,0);
            for(i=0;i<n;i++)
            {
                for(j=i+1;j<n;j++)
if((cir[i].x-cir[j].x)*(cir[i].x-cir[j].x)+(cir[i].y-cir[j].y)*(cir[i].y-cir[j].y)<(cir[i].r+cir[j].r)*(cir[i].r+cir[j].r))
                        {
                            GetPoint(i,j,p1,p2);
```

```
                    for(k=0;k<n;k++)
                        if(k!=i&&k!=j&&Cover(k,p1))break;
                    if(k==n)
                    {
p[i][nsec[i]].x=p1.x;p[i][nsec[i]].y=p1.y;
p[j][nsec[j]].x=p1.x;p[j][nsec[j]].y=p1.y;
p[i][nsec[i]].x0=cir[i].x;p[i][nsec[i]].y0=cir[i].y;
p[j][nsec[j]].x0=cir[j].x;p[j][nsec[j]].y0=cir[j].y;
p[i][nsec[i]].other=j*100+nsec[j];
p[j][nsec[j]].other=i*100+nsec[i];
if((p1.x-cir[i].x)*(p2.y-cir[i].y)-(p1.y-cir[i].y)*(p2.x-cir[i].x)<-NEG) {
p[i][nsec[i]].sign='+';p[j][nsec[j]].sign='-';
        }
        else{
p[i][nsec[i]].sign='-';p[j][nsec[j]].sign='+';
          }
          nsec[i]++;nsec[j]++;
   }
     for(k=0;k<n;k++)
        if(k!=i&&k!=j&&Cover(k,p2))break;
        if(k==n)
        {
p[i][nsec[i]].x=p2.x;p[i][nsec[i]].y=p2.y;
```

```
p[j][nsec[j]].x=p2.x;p[j][nsec[j]].y=p2.y;
p[i][nsec[i]].x0=cir[i].x;p[i][nsec[i]].y0=cir[i].y;
p[j][nsec[j]].x0=cir[j].x;p[j][nsec[j]].y0=cir[j].y;
p[i][nsec[i]].other=j*100+nsec[j];
p[j][nsec[j]].other=i*100+nsec[i];
if((p1.x-cir[i].x)*(p2.y-cir[i].y)-(p1.y-cir[i].y)*(p2.x-cir[i].x)<-NEG) {
p[i][nsec[i]].sign='-';p[j][nsec[j]].sign='+';
                }
                else{
p[i][nsec[i]].sign='+';p[j][nsec[j]].sign='-';
            }
                                nsec[i]++;nsec[j]++;
                        }
                    }
                quicksort(p[i],0,nsec[i]-1,i);
            }
            fill(&v[0][0],&v[n][0],false);
            int s=0;
            for(i=0;i<n;i++)
                for(j=0;j<nsec[i];j++)
                    if(!v[i][j])
                        if(p[i][j].sign=='-')s+=Visit(i,j);
                        else s+=Visit(p[i][j].other/100,p[i][j].other%100);
```

```
        cout<<s<<endl;
    }
    return 0;
}
```

1427、An Old Stone Game

```
#include<iostream.h>
#include<stdlib.h>
typedef struct
{
        int r; //所需的石子数
        int son; //指向儿子节点的指针
}NODE;  //节点类型
int sons[202];  //所有节点的儿子
NODE p[201];//保存节点
int comp(const void *e1,const void *e2)
{
        return *(int *)e2 - *(int *)e1;
}
int done(int k)
{

        if(p[k].r != -1)
                return p[k].r;
```

```
        int u,v,j;
        int a[200];
        v = p[k].son;
        u = sons[v++];
        for(j=0;j<u;j++)
        {
                if(p[sons[j+v]].r==-1)
                        p[sons[j+v]].r = done(sons[j+v]);//求 k 的子节点 sons[j+v]所需石子数
                a[j] = p[sons[j+v]].r;
        }
        qsort(a,u,sizeof(int),comp);////对 a[]快速排序
        int max = 0;
        for(j=0;j<u;j++)
                if(max < a[j] + j)
                        max = a[j] + j;
        return max;
}
int main()
{

        int t,n;
        int i,j,k,m;
        cin>>t;
```

```
while(t-- >0)
{
        cin>>n;
        int sn = 0;
        for(i=0;i<n;i++)
        {
                cin>>k>>m;
                if(m==0)
                {
                        p[k].r = 1;
                        p[k].son = -1;
                        continue;
                }
                p[k].son = sn;p[k].r = -1;
                sons[sn++] = m;
                for(j=0;j<m;j++)
                        cin>>sons[j+sn]; // 读入节点 k 的所有子节点
                sn += m;
        }
        j = done(1);//取根节点所需石子数
        cout<<j<<endl;
}
return 0;
```

```
}
```

1428、Magazine Delivery

```
#include<iostream.h>
#include<memory.h>
int m,n,graph[30][30],dis[30][30][30] ;
int D(int a,int b,int c )
{
        int tmp,d1,d2,d3 ;
        if ( c == m ) return 0;
    if ( dis[b][c][c+1] == 0 ) dis[b][c][c+1] = D(b,c,c+1) ;
        if ( dis[a][b][c+1] == 0 ) dis[a][b][c+1] = D(a,b,c+1) ;
        if ( dis[a][c][c+1] == 0 ) dis[a][c][c+1] = D(a,c,c+1) ;
        d1 = dis[b][c][c+1] + graph[a][c+1] ;
        d2 = dis[a][b][c+1] + graph[c][c+1] ;
        d3 = dis[a][c][c+1] + graph[b][c+1] ;
    if ( d2 > d1 ) tmp = d1;
        else tmp = d2;
        if ( tmp > d3 ) tmp = d3 ;
        return tmp;
}
int main()
{
        int i,j,v;
```

```
        cin >> n ;
        while(n--)
        {
                cin >> m;
                memset( graph,0,sizeof(graph));
                for( i = 0 ; i < m-1; i++ )
                {
                        for( j = i+1; j < m ; j++ )
                        {
                        cin >> graph[i][j] ;
                        graph[j][i] = graph[i][j];
                        }
                }
                memset( dis,0,sizeof(dis)) ;
                cout << D(0,0,0) << endl;
        }
        return 0;
}
```

1430、The Erythea Campaign

```
#include <iostream.h>
#include <string.h>
const int MAXN = 100;
const int MAXV = 100000000;
```

```
int danger[MAXN][MAXN], map[MAXN][MAXN], rc[MAXN * MAXN * 2], dis[MAXN*MAXN]
        , pos[MAXN*MAXN];
int m, n, s0, t0, s, t;
void calcdanger()
{
        int i, j, x, y;

        t = 0;
        memset(danger, 0, sizeof(danger));
        for (i = 0; i <= n; i++)
                for (j = 0; j <= m; j++)
                if (map[i][j] || map[i+1][j] || map[i][j+1] || map[i+1][j+1])
                {
                        danger[i][j] = 1;
                        rc[++t] = i * (m+1) + j;
                }
        s = 1;
        while (s <= t)
        {
                x = rc[s] / (m+1); y = rc[s] % (m+1);
                if (x > 0 && !danger[x-1][y])
                {
                danger[x-1][y] = danger[x][y]+1;
```

```
                rc[++t] = rc[s]-m-1;
            }
            if (y > 0 && !danger[x][y-1])
            {
            danger[x][y-1] = danger[x][y]+1;
                rc[++t] = rc[s]-1;
            }
            if (x < n && !danger[x+1][y])
            {
            danger[x+1][y] = danger[x][y]+1;
                rc[++t] = rc[s]+m+1;
            }
            if (y < m && !danger[x][y+1])
            {
            danger[x][y+1] = danger[x][y]+1;
                rc[++t] = rc[s]+1;
            }
            s++;
        }
        for (i = 0; i <= n; i++)
            for (j = 0; j <= n; j++)
            danger[i][j] = m+n+1-danger[i][j];
}
```

```
int getHead()
{
	int a, i, j, x;
	a = rc[1];
	x = rc[t--];
	i = 1;
	while (i * 2 <= t)
	{
		j = i * 2;
		if (j + 1 <= t && dis[rc[j]] > dis[rc[j+1]]) j = j + 1;
		if (dis[rc[j]] < dis[x])
		{
			rc[i] = rc[j]; pos[rc[i]] = i;
			i = j;
		}
		else break;
	}
	rc[i] = x; pos[x] = i;
	return a;
}
void add(int node)
{
	int x;
```

```
        if (pos[node] == 0)
        {
                rc[++t] = node; pos[node] = t;
        }
        x = pos[node];
        while (x > 1)
        {
                if (dis[rc[x/2]] > dis[node])
                {
                        rc[x] = rc[x/2]; pos[rc[x]] = x;
                        x = x / 2;
                }
                else break;
        }
        rc[x] = node; pos[node] = x;
}
void calcdis()
{
        int i, a, x, y;
        memset(pos, 0, sizeof(pos));
      for (i = 0; i <= (m+1)*(n+1); i++) dis[i] = MAXV;
        dis[s0] = danger[s0/(m+1)][s0%(m+1)];
        rc[t = 1] = s0; pos[s0] = 1;
```

```
        while (t > 0)
        {
                a = getHead();
                if (a == t0) break;
                x = a/(m+1); y = a%(m+1);

if (x > 0 && (!map[x][y] || !map[x][y+1])
                        && dis[a-m-1] > danger[x-1][y]+dis[a])
                {
                dis[a-m-1] = danger[x-1][y]+dis[a];
                        add(a-m-1);
                }
                if (y > 0 && (!map[x][y] || !map[x+1][y])
               && dis[a-1] > danger[x][y-1]+dis[a])
                {
                dis[a-1] = danger[x][y-1]+dis[a];
                        add(a-1);
                }
                if (x < n && (!map[x+1][y] || !map[x+1][y+1])
        && dis[a+m+1] > danger[x+1][y]+dis[a])
                {
              dis[a+m+1] = danger[x+1][y]+dis[a];
                        add(a+m+1);
```

```
                }
                if (y < m && (!map[x][y+1] || !map[x+1][y+1])
            && dis[a+1] > danger[x][y+1]+dis[a])
                {
                dis[a+1] = danger[x][y+1]+dis[a];
                        add(a+1);
                }
        }
        if (dis[t0] < MAXV) cout << dis[t0] << endl;
                else cout << "no solution" << endl;
}
int main()
{
        int i, j, test, c, r;
        char ch;
        cin >> test;
        while (test-- > 0)
        {
                cin >> n >> m;
                cin >> c >> r;
                s0 = c * (m + 1) + r;
                cin >> c >> r;
                t0 = c * (m + 1) + r;
```

```
                memset(map, 0, sizeof(map));
                for (i = 1; i <= n; i++)
                        for (j = 1; j <= m; j++)
                        {
                                cin >> ch;
                                map[i][j] = ch - '0';
                        }
                calcdanger();
                calcdis();
        }
        return 0;
}
```

1519、 Will Indiana Jones Get There

```
#include<iostream.h>
#include<math.h>
#include<iomanip.h>
struct point// 一个点
{
   int x,y;
};
struct
{
        point begin,end;
```

```
    int sign;
}edge[1000];//一条边
double calculate(point a,point b) //计算两个点之间的距离
{
        return (sqrt((a.x-b.x+0.0)*1.0*(a.x-b.x)+(a.y-b.y+0.0)*1.0*(a.y-b.y)));
}
double distance(int i,int last)//计算两条线之间的距离
{
        int erect,flat;
        double between;
    if(edge[i].sign==edge[last].sign)
        {
            if(edge[i].sign==1) //两条线段都是水平的
            {
        if(edge[i].end.x<edge[last].begin.x) between=calculate(edge[i].end,edge[last].begin);
                                else if(edge[i].begin.x>edge[last].end.x) between=calculate(edge[i].begin,edge[last].end);
                                    else between=fabs(edge[i].begin.y-edge[last].begin.y);
            }
            else //两条线段都是竖直的
            {
        if(edge[i].begin.y>edge[last].end.y) between=calculate(edge[i].begin,edge[last].end);
                                else if(edge[i].end.y<edge[last].begin.y) between=calculate(edge[i].end,edge[last].begin);
                                    else between=fabs(edge[i].begin.x-edge[last].begin.x);
```

```
            }
}
else
{
             if(edge[i].sign==1) {erect=last;flat=i;}
             else {erect=i;flat=last;} //erect 记录竖线，flat 记录横线
             if(edge[erect].end.x<edge[flat].begin.x) //图 4，5，6 中，竖线均在横线的左侧。
             {
if(edge[erect].begin.y>edge[flat].begin.y)
between=calculate(edge[erect].begin,edge[flat].begin);
                        else if(edge[erect].end.y<edge[flat].begin.y)
between=calculate(edge[erect].end,edge[flat].begin);
                        else between=fabs(edge[erect].begin.x-edge[flat].begin.x);
            }
            else if(edge[erect].begin.x>edge[flat].end.x)//图 7，8，9 中，竖线均在横线的右侧
            {
if(edge[erect].begin.y>edge[flat].begin.y)
between=calculate(edge[erect].begin,edge[flat].end);
                        else if(edge[erect].end.y<edge[flat].begin.y)
                                between=calculate(edge[erect].end,edge[flat].end);
                        else between=fabs(edge[erect].begin.x-edge[flat].end.x);
            }
            else //图 10，图 11，图 12 中，竖线的 x 坐标介于横线两端的 x 坐标之间
```

```
                {
        if(edge[erect].begin.y>edge[flat].begin.y)
        between=fabs(edge[erect].begin.y-edge[flat].begin.y);
                        else if(edge[erect].end.y<edge[flat].begin.y)
                                between=fabs(edge[erect].end.y-edge[flat].begin.y);
                        else between=0;
                }
        }
    return between;
}
int main()
{
        int i,n,l,min,used[1000],last;
        double between,weight[1000];
        cin>>n;
        while(n>0)
        {
                for(i=0;i<n;i++)//初始化，记录各条边
                {
        cin>>edge[i].begin.x>>edge[i].begin.y>>l;
                        if(l>0)
                        {
        edge[i].end.x=edge[i].begin.x+l;
```

```
edge[i].end.y=edge[i].begin.y;
                    edge[i].sign=1;//1 表示 weihengxian
                }
                else
                {
edge[i].end.x=edge[i].begin.x;
edge[i].end.y=edge[i].begin.y-l;
edge[i].sign=-1;// -1 表示为竖线
                }
            }
            for(i=0;i<n;i++)
            {
                weight[i]=9999999;
            used[i]=0;// 0 表示未被标记过
            }
            last=0;
        used[0]=1;//标记起始点已被选过
            weight[0]=0;                while(last!=1)
            {
                for(i=0;i<n;i++)
                    if(used[i]==0)
                    {
between=distance(i,last);//求 i 到 last 的距离
```

```
if(weight[last]>between) between=weight[last];
if((weight[i]-between)>1e-10) weight[i]=between;
                                                    }
                                                    i=0;
            while(used[i]==1) i++;
                                                    min=i;
                                                    for(i;i<n;i++)
      if((used[i]==0)&&((weight[min]-weight[i])>1e-10))
                                                            min=i;
                    last=min;
                                                    used[min]=1;
                      }
            cout<<setiosflags(ios::fixed)<<setprecision(2)<<weight[1]<<endl;
                      cin>>n;
            }
            return 0;
}
```

1734、Power Network

```
#include <stdio.h>
#include <memory.h>
int n, np, nc, m, s, t;
int fa[104], q[104], f[104][104], c[104][104];
int abs(int x)
```

```
{
    return x>0? x:-x;
}
void proc()
{
    int qs, qt, d, d0, i, j, ans = 0;
    fa[t] = 1;
    while (fa[t] != 0)
    {
        qs = 0; qt = 1;                                    //队列的首尾指针初始化
        q[qt] = s;
        memset(fa, 0, sizeof(fa));                         //增广路径初始化
        fa[s] = s;
        while (qs < qt && fa[t] == 0)                      //若没有找到到汇点的增广路或还可以继续寻找增广路
        {
            i = q[++qs];
            for (j = 1; j <= t; j++)
                if (fa[j] == 0)                            //点 j 没有标记过
                  if (f[i][j] < c[i][j])
                  {
                  fa[j] = i;
                  q[++qt] = j;
```

```
                }
                else
                    if (f[j][i] > 0)
                    {
                fa[j] = -i;
                q[++qt] = j;
                }
        }
        if (fa[t] != 0)                                    //如果找到一条从源点
到汇点的增广路就改进当前流
        {
                d0 = 10000000;
                int i;
                i = t;
                while (i != s)                             //寻找最大的可改进量
                {
                        if (fa[i] > 0)
                        {
        if ((d = c[fa[i]][i] - f[fa[i]][i]) < d0)
                d0 = d;
                }
                        else
                    if (f[i][-fa[i]] < d0)
                        d0 = f[i][-fa[i]];
```

```
                i = abs(fa[i]);
            }
            ans += d0;                                          //总流量累加
            i = t;
            while (i != s)                                      //改进流
            {
        if (fa[i] > 0)
        f[fa[i]][i] += d0;
            else
        f[i][-fa[i]] -= d0;
        i = (int)abs(fa[i]);
            }
        }
        printf("%d\n", ans);                                    //输出最大流
}
int main()
{
        int i, u, v, cc;
while (scanf("%d%d%d%d", &n, &np, &nc, &m) == 4)
        {
            s = n + 2; t = n + 1;                               //以下是构图
            memset(f, 0, sizeof(f));
```

```
            memset(c, 0, sizeof(c));
            for (i = 1; i <= m; i++)                    //对于原图中边(u,v)连一条容量为 cc 的弧
            {
                while (getchar() != '(');
            scanf("%d,%d)%d", &u, &v, &cc);
                c[u + 1][v + 1] = cc;
            }
            for (i = 1; i <= np; i++)                   //对于 PowerStation 从源点连一条容量为 cc 的弧
            {
                while (getchar() != '(');
              scanf("%d)%d", &u, &cc);
                c[s][u + 1] = cc;
            }
            for (i = 1; i <= nc; i++)                   //对于 Consumer 连一条容量为 cc 的弧到汇点
            {
                while (getchar() != '(');
               scanf("%d)%d", &u, &cc);
                c[u + 1][t] = cc;
            }
            proc();                                                          //求最大流
        }
        return 0;
}
```

1812、Stamps

```
#include<iostream.h>
#include<search.h>
#include<memory.h>
const int MAX_STAMP_TYPES =100;
const int MAX_TOT_VALUE = 100;
const int MAX_CUSTOMER=100;
int s[MAX_STAMP_TYPES],s_size,cus[MAX_CUSTOMER],c_size,maxc,count[100];
bool custom[MAX_TOT_VALUE];
class Combine{
public:
    int tot;
    int ts;//types of stamps
    int f;//flag 1 can 2 tie,0 can't
    int s[4];//each stamps used here
    int sc;//used stamps number
    int max;//single maxed stamp value;
public:
    bool isHave(int type){
        for(int i=0;i<sc;i++)
            if(s[i]==type)return true;
        return false;
    }
```

```
    void findMax(){
        max=0;
        for(int i=0;i<sc;i++)
            if(::s[s[i]]>max)max=::s[s[i]];
    }
    void addStamp(int type){
        if(!isHave(type))ts++;
        s[sc++]=type;
        tot+=::s[type];
        if(::s[type]>max)max=::s[type];
    }
    void removeLastStamp(int type){
        sc--;
        if(!isHave(type))ts--;
        tot-=::s[type];
        findMax();
    }
}currCus[MAX_TOT_VALUE],c;
int com(const void * a,const void *b)
{
    return *((int *)a)-*((int*)b);
}
void init()
```

```
{
    maxc=c_size=s_size=0;
    memset(custom,0,MAX_TOT_VALUE*sizeof(bool));
    memset(count,0,100*sizeof(int));
}
bool read()
{
    init();
    int t;
    cin>>t;
    if(!cin)return false;
    while(t!=0){
        if(count[t]++<5)s[s_size++]=t; //modify by duoshute here
        cin>>t;
    }
    qsort(s,s_size,sizeof(int),com);
    cin>>t;
    while(t!=0){
        cus[c_size++]=t;
        custom[t]=true;
        if(t>maxc)maxc=t;
        cin>>t;
    }
```

```
        return true;
}
int a[4],b[4];
bool same(int *t1,int *t2,int n)
{


        memcpy(a,t1,n*sizeof(int));
        memcpy(b,t2,n*sizeof(int));
/*      qsort(a,n,sizeof(int),com);                     //omited by duoshute
        qsort(b,n,sizeof(int),com); */
        return memcmp(a,b,n*sizeof(int))==0;
}
void solve(int ss,int last) //int last is added by duoshute
{
        if(custom[c.tot])
        {
                if(!currCus[c.tot].f
                                ||c.ts>currCus[c.tot].ts
                                ||c.ts==currCus[c.tot].ts&&c.sc<currCus[c.tot].sc
                                ||c.ts==currCus[c.tot].ts&&c.sc==currCus[c.tot].sc&&c.max>currCus[c.tot].max)
                {
                        currCus[c.tot]=c;
                        currCus[c.tot].f=1;
```

```
            }
            else if(c.ts==currCus[c.tot].ts&&c.sc==currCus[c.tot].sc&&c.max==currCus[c.tot].max)
            {
                if(!same(c.s,currCus[c.tot].s,c.sc))
                    currCus[c.tot].f=2;
            }

        }
        if(c.sc>=4||c.tot>maxc)return;
        for(int i=last;i<s_size;i++)
        {
            c.addStamp(i);
            solve(ss+1,i);
            c.removeLastStamp(i);
        }
}
int main()
{
        while(read())
        {
            memset(&c,0,sizeof(Combine));
            memset(currCus,0,MAX_TOT_VALUE*sizeof(Combine));
            solve(0,0);
```

```
        for(int i=0;i<c_size;i++)
            if(currCus[cus[i]].f==0)
                cout<<cus[i]<<" ---- none"<<endl;
            else if(currCus[cus[i]].f==2)
                cout<<cus[i]<<" ("<<currCus[cus[i]].ts<<"): "<<"tie"<<endl;
            else
            {
                cout<<cus[i]<<" ("<<currCus[cus[i]].ts<<"):";
                qsort(currCus[cus[i]].s,currCus[cus[i]].sc,sizeof(int),com);
                for(int j=0;j<currCus[cus[i]].sc;j++)
                    cout<<' '<<s[currCus[cus[i]].s[j]];
                cout<<endl;
            }
    }
    return 0;
}
```

2239、In Danger

```
#include <iostream>
#include <vector>
using namespace std;
vector<int> mode;
int main()
{
```

```
double buf;
while(cin>>buf)
{
    int n=int(buf);
    if (n==0) break;
    int cur_mode=2;
    int past_mode;
    mode.clear();
    while(n>1)
    {
        mode.push_back(cur_mode);
        past_mode=cur_mode;
        if (n%2)
        {
            if (cur_mode==2)
            {
                cur_mode=1;
            }
            else
            {
                cur_mode=2;
            }
        }
```

```
            if (past_mode==2)
            {
                if (n%2) n=n/2+1;
                else n/=2;
            }
            else
            {
                n/=2;
            }
        }
        int result=1;
        vector<int>::reverse_iterator pmode;
        for (pmode=mode.rbegin();pmode!=mode.rend();pmode++)
        {
            result*=2;
            if (*pmode==2) result-=1;
        }
        cout<<result<<endl;
    }
}
```

1938、Binomial Showdown

```
#include<iostream.h>
using namespace std;
```

```
int main()
{
    int n,k;
    double temp;
    while(cin>>n>>k)
    {
        if(n==0&&k==0) break;
        else
        {
            temp=1;
            if(n-k>k)
            {
                for(int i=n-k+1;i<=n;++i)
                    temp*=i;
                for(int j=1;j<=k;++j)
                    temp/=j;
                cout<<(long)temp<<endl;                }
            else
            {
                for(int i=k+1;i<=n;++i)
                    temp*=i;
                for(int j=1;j<=n-k;++j)
                    temp/=j;
```

```
                cout<<(long)temp<<endl;                 }
        }
    }
    return 0;
}
```

2420、Calendar

```
#include<iostream.h>
#include<stdlib.h>
#include<stdio.h>
using namespace std;
int mday[2][13] = {   0,31,28,31,30,31,30,31,31,30,31,30,31,0,31,29,31,30,31,30,31,31,30,31,30,31 };
char week[7][20] = { "Saturday","Sunday","Monday","Tuesday","Wednesday","Thursday","Friday" };
int main ()
{
    int n;
    int s, y, m, d, t;
    while ( 1 ) {
        cin >> n ;
        if ( n < 0 )
            break;
        d = n % 7;
        n ++ ;
        for ( y = s = 0 ; s < n; s += t, y ++ )
```

```
        if ( y % 400 == 0 || y % 4 == 0 && y % 100 )
                t = 366;
            else
                t = 365;
            y -- ; s -= t; n -= s;
            cout << y+2000 << "-";
            t -= 365;
    for ( m = 0, s = 0; s < n; m ++,s += mday[t][m]    );
            {
                if(m<10)
                   cout<<"0"<<m<<"-";
                else
                   cout<<m<<"-";
            }
            s -= mday[t][m]; n -= s;
                if(n<10)
                   cout<<"0"<<n<<" ";
                else
                   cout<<n<<" ";
        cout << week[d] << endl;
    }
    return 0;
}
```

附一（常用排序方法）

- Shell 排序

  Shell 排序是以发明者命名的一种较快的排序方法。Shell 排序基本算法思想是：将整个无序序列分割成若干小的子序分别进行插入排序。

  子序列的分割方法为：将相隔某个增量 h 的元素构成一个子序列。在排序过程中，逐次减小这个增量，?最后当 h 减到 1 时，进行一次插入排序，排序就完成。

  在本函数中，增量序列取 ht=2t－1，1 tlog2n 其中 n 为待排序序列的长度。

  例：（/＊ 将输入的数据排序后,输出一个测试 Shell 排序的主函数＊/）

```
＃define SIZE 10
main() {
void shell();
int d[SIZE],i;
printf(“Input %d numbers\n″,SIZE);
for(i=0;i
scanf(“%d″,＆d[i]);
shell(d,SIZE);
printf(“After sort:\n″)
for(i=0;i
printf(“%5d″,d[i]);
printf(“\n″);}
/＊把数组 V 的元素按增序排序＊/
void shell(v,n)
int v[],n;
{int gap,i,j,temp;
for(gap=n/2;gap>0;gap/=2)
for(i=gap;i
for(j=i－gap;j>=0 ＆＆ v[j])
v[j＋gap];j－=gap)
{ temp=v[j];
v[j]=v[j＋gap];
v[j＋gap]=temp; }
```

  注：这里，数组作为函数参数，参数组中元素值的改变就会反过来影响到实参数组。

- 选择排序

  选择排序基本算法思想：首先找出最小的元素，然后把这个元素与第一个元素互换，这样值最小的元素就放到了第一个位置；接着，再从剩下的元素中找值最小的，把它和第二个元素互换，使得第二小的元素放在第二个位置上，依此类推，直到所有的值由小到大排列为止。

  例： # define NUM 10

```
main()
{int a[NUM],i,j,r,temp;
printf(“Please input %d number\n″,NUM)
for (i=0;i
scanf(″%d″,＆a[i]);
for(i=0;i
r=i;
```

```
for(i=i＋1;j
if(a[i]
r=j;
if(r!=i)
temp=a[i];a[i]=a[r];a[r]=temp;} }
printf("The array after sort:\n")
for(i=0;i
printf("%5d",a[i]);
printf("\n"); }
```

- 快速排序

快速排序是目前使用较好的排序算法，它是由 C.A.Hoare 发明并命名的。快速排序基本算法思想：通过一次分割，将无序序列分成两部分，其中前一部分的元素值均不大于后一部分的元素值。然后对每一部分利用同样的方法进行分割，这个过程一直做到每一个子序列的长度小于某个值 m 为止。

对序列 p 的分割过程： 首先，在序列的第一个、中间一个及最后一个元素中选取中项，得 p（k），并将 p（k）赋给 t；再将序列中的第一个元素移到 p（k）的位置上；然后设置两个指针 i 和 j 分别指向序列的起始和最后的位置。

```
例： void quick(v,n)
int v[],n;
{ void qs();
qs(v,0,n－1);
/＊快速排序，数组方案＊/
void qs(v,left,right)
int v[],left,ringt;
{ int i,j,x,temp;
i=left;
v=v[(left＋right)/2];
while (i
while([i]
j－－;
if(i<=j){
temp=v[i];
v[i]=v[j];
v[j]=temp;
i＋＋;
j－－; } }
if (left
qs(v,left,j);
if(i
qs(v,i,right); }
```

注：在这个递归函数例子中，数组 V 既做为形参数，又做为实际参数。

- 冒泡排序

冒泡排序基本算法思想：从前到后扫描序列，比较相邻两个项目的大小，若发现逆序进行交换，最后使最大者换到序列的最后；然后再从后到前扫描剩下的序列，比较相邻两个项目的大小，若发现逆序则进行交换，最后使最小者换到序列的最前面。对剩下的序列重复上述过程，直到剩下的序列为空止。

```
例：void ma(p,n)
int P[],n;
{ int m,k,j,i,d;
```

```
    k=0;m=n－1;
    while (k
    { j=m－1;m=0;
    for(ik;i<=;i＋＋)
    if(p[i]>p[i＋1])
    { d=p[i];p[i]=p[i＋1];
    p[i＋1]=d;m=i;}
    j=k＋1;k=0;
    for(i=m;i>=j;i－－)
    if (p[i－1]>p[i])
    {d=p[i－1];p[i]=p[i－1];
    p[i－1]=d;k=i;} }
    return; }
```

排序举例

```
/*对键盘输入的十个数从小到大排序*/
/*冒泡排序（buddle  sort）*/
main()
{
int  a[11];
int  i,j,t;
clrscr();
printf("input  10  numbers  :\n");
for(i=1;i<11;i++)
    scanf("%d",&a[i]);
    printf("\n");
    for(j=1;j<=9;j++)
        for(i=1;i<=10-j;i++)
        if(a[i]>a[i+1])
            {t=a[i];a[i]=a[i+1];a[i+1]=t;}
        printf("the  sorted  numbers:\n");
        for(i=1;i<11;i++)
        printf("%d  ",a[i]);
        getch();
        }
/*选择排序(select  sort)*/
main()
{
int i,j,a[10],t;

for(i=0;i<=9;i++)
scanf("%d",&a[i]);
for(i=0;i<=8;i++)
for(j=i;j<=8;j++)
{t=a[i];
  if(a[j]>a[j+1])
  {a[i]=a[j+1];
```

```
    a[j+1]=t;}
}
for(i=0;i<=9;i++)
printf("%d  ",a[i]);
}
/*插入法排序*/
#define  N  10
#include"stdio.h"
main()
{  int  i,j,k,t,a[N];
clrscr();
printf("Please  input  %d  numbers:\n",N);
for(i=0;i<N;i++)
scanf("%d",&a[i]);
  for(i=1;i<N;i++)
  {
    for(j=0;j<i;j++)
    {if(a[j]>a[i])
      {t=a[i];
        for(k=i;k>=j;k--)
        a[k]=a[k-1];
        a[j]=t;
      }
    }   }
  printf("small  to  big  order:\n");
  for(i=0;i<N;i++)
  printf("%-2d",a[i]);
  printf("\n");
  getch(); }
```

ASCII 码表

Table of ASCII CharactersThis table lists the ASCII characters and their decimal, octal and hexadecimal numbers. Characters which appear as names in parentheses (e.g., (nl)) are non-printing characters. A table of the common non-printing characters appears after this table.

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(nul)	0	0000	0x00	(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
(soh)	1	0001	0x01	!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
(stx)	2	0002	0x02	″	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
(etx)	3	0003	0x03	#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
(eot)	4	0004	0x04	$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
(enq)	5	0005	0x05	%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
(ack)	6	0006	0x06	&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
(bel)	7	0007	0x07	’	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(bs)	8	0010	0x08	(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
(ht)	9	0011	0x09	)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
(nl)	10	0012	0x0a	*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
(vt)	11	0013	0x0b	+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
(np)	12	0014	0x0c	,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
(cr)	13	0015	0x0d	-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
(so)	14	0016	0x0e	.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
(si)	15	0017	0x0f	/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
(dle)	16	0020	0x10	0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
(dc1)	17	0021	0x11	1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
(dc2)	18	0022	0x12	2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72

(dc3) 19 0023 0x13 | 3 51 0063 0x33 | S 83 0123 0x53 | s 115 0163 0x73
(dc4) 20 0024 0x14 | 4 52 0064 0x34 | T 84 0124 0x54 | t 116 0164 0x74
(nak) 21 0025 0x15 | 5 53 0065 0x35 | U 85 0125 0x55 | u 117 0165 0x75
(syn) 22 0026 0x16 | 6 54 0066 0x36 | V 86 0126 0x56 | v 118 0166 0x76
(etb) 23 0027 0x17 | 7 55 0067 0x37 | W 87 0127 0x57 | w 119 0167 0x77
(can) 24 0030 0x18 | 8 56 0070 0x38 | X 88 0130 0x58 | x 120 0170 0x78
(em) 25 0031 0x19 | 9 57 0071 0x39 | Y 89 0131 0x59 | y 121 0171 0x79
(sub) 26 0032 0x1a | : 58 0072 0x3a | Z 90 0132 0x5a | z 122 0172 0x7a
(esc) 27 0033 0x1b | ; 59 0073 0x3b | [91 0133 0x5b | { 123 0173 0x7b
(fs) 28 0034 0x1c | < 60 0074 0x3c | \ 92 0134 0x5c | | 124 0174 0x7c
(gs) 29 0035 0x1d | = 61 0075 0x3d |] 93 0135 0x5d | } 125 0175 0x7d
(rs) 30 0036 0x1e | > 62 0076 0x3e | ^ 94 0136 0x5e | ~ 126 0176 0x7e
(us) 31 0037 0x1f | ? 63 0077 0x3f | _ 95 0137 0x5f | (del) 127 0177 0x7f

ASCII Name Description C Escape Sequence nul null byte \0 bel bell character \a bs backspace \b ht horizontal tab \t np formfeed \f nl newline \n cr carriage return \r vt vertical tab esc escape sp space

Turbo C 2.0 部分函数中文说明大全

数学函数,所在函数库为 math.h、stdlib.h、string.h、float.h

int abs(int i) //返回整型参数 i 的绝对值

double fabs(double x) //返回双精度参数 x 的绝对值

long labs(long n) //返回长整型参数 n 的绝对值

double exp(double x) //返回指数函数 ex 的值

double frexp(double value,int *eptr) //返回 value=x*2n 中 x 的值,n 存贮在 eptr 中

double ldexp(double value,int exp); //返回 value*2exp 的值

double log(double x) 返回 logex 的值

double log10(double x) 返回 log10x 的值

double pow(double x,double y) 返回 xy 的值

double pow10(int p) 返回 10p 的值

double sqrt(double x) 返回 x 的开方

double acos(double x) 返回 x 的反余弦 cos-1(x)值,x 为弧度

double asin(double x) 返回 x 的反正弦 sin-1(x)值,x 为弧度

double atan(double x) 返回 x 的反正切 tan-1(x)值,x 为弧度

double atan2(double y,double x) 返回 y/x 的反正切 tan-1(x)值,y 的 x 为弧度

double cos(double x) 返回 x 的余弦 cos(x)值,x 为弧度

double sin(double x) 返回 x 的正弦 sin(x)值,x 为弧度

double tan(double x) 返回 x 的正切 tan(x)值,x 为弧度

double cosh(double x) 返回 x 的双曲余弦 cosh(x)值,x 为弧度

double sinh(double x) 返回 x 的双曲正弦 sinh(x)值,x 为弧度

double tanh(double x) 返回 x 的双曲正切 tanh(x)值,x 为弧度

double hypot(double x,double y) 返回直角三角形斜边的长度(z), x 和 y 为直角边的长度,z2=x2+y2

double ceil(double x) 返回不小于 x 的最小整数

double floor(double x) 返回不大于 x 的最大整数

void srand(unsigned seed) 初始化随机数发生器

int rand() 产生一个随机数并返回这个数

double poly(double x,int n,double c[]) 从参数产生一个多项式

double modf(double value,double *iptr) 将双精度数 value 分解成尾数和阶

double fmod(double x,double y) 返回 x/y 的余数

类型转换

double atof(char *nptr) 将字符串 nptr 转换成浮点数并返回这个浮点数

double atoi(char *nptr) 将字符串 nptr 转换成整数并返回这个整数

double atol(char *nptr) 将字符串 nptr 转换成长整数并返回这个整数

char *ecvt(double value,int ndigit,int *decpt,int *sign) 将浮点数 value 转换成字符串并返回该字符串

char *fcvt(double value,int ndigit,int *decpt,int *sign) 将浮点数 value 转换成字符串并返回该字符串

char *gcvt(double value,int ndigit,char *buf) 将数 value 转换成字符串并存于 buf 中,并返回 buf 的指针

char *ultoa(unsigned long value,char *string,int radix) 将无符号整型数 value 转换成字符串并返回该字符串,radix 为转换时所用基数

char *ltoa(long value,char *string,int radix) 将长整型数 value 转换成字符串并返回该字符串,radix 为转换时所用基数

char *itoa(int value,char *string,int radix) 将整数 value 转换成字符串存入 string,radix 为转换时所用基数

double atof(char *nptr) 将字符串 nptr 转换成双精度数,并返回这个数,错误返回 0

int atoi(char *nptr) 将字符串 nptr 转换成整型数, 并返回这个数,错误返回 0

long atol(char *nptr) 将字符串 nptr 转换成长整型数,并返回这个数,错误返回 0

double strtod(char *str,char **endptr)将字符串 str 转换成双精度数,并返回这个数,

long strtol(char *str,char **endptr,int base)将字符串 str 转换成长整型数, 并返回这个数,

int matherr(struct exception *e) 用户修改数学错误返回信息函数(没有必要使用)

unsigned int _clear87() 清除浮点状态字并返回原来的浮点状态

char *ecvt(double value,int ndigit,int *decpt,int *sign) 将浮点数 value 转换成字符串并返回该字符串

char *fcvt(double value,int ndigit,int *decpt,int *sign) 将浮点数 value 转换成字符串并返回该字符串

char *gcvt(double value,int ndigit,char *buf) 将数 value 转换成字符串并存于 buf 中,并返回 buf 的指针

char *ultoa(unsigned long value,char *string,int radix) 将无符号整型数 value 转换成字符串并返回该字符串,radix 为转换时所用基数

char *ltoa(long value,char *string,int radix) 将长整型数 value 转换成字符串并返回该字符串,radix 为转换时所用基数

char *itoa(int value,char *string,int radix) 将整数 value 转换成字符串存入 string,radix 为转换时所用基数

double atof(char *nptr) 将字符串 nptr 转换成双精度数,并返回这个数,错误返回 0

int atoi(char *nptr) 将字符串 nptr 转换成整型数, 并返回这个数,错误返回 0

long atol(char *nptr) 将字符串 nptr 转换成长整型数,并返回这个数,错误返回 0

double strtod(char *str,char **endptr) 将字符串 str 转换成双精度数,并返回这个数,

long strtol(char *str,char **endptr,int base) 将字符串 str 转换成长整型数, 并返回这个数,

int toascii(int c) 返回 c 相应的 ASCII

int tolower(int ch) 若 ch 是大写字母('A'-'Z')返回相应的小写字母('a'- 'z')

int _tolower(int ch) 返回 ch 相应的小写字母('a'-'z')

int toupper(int ch) 若 ch 是小写字母('a'-'z')返回相应的大写字母('A'- 'Z')

int _toupper(int ch) 返回 ch 相应的大写字母('A'-'Z')

str...字符串操作函数

char stpcpy(char *dest,const char *src) 将字符串 src 复制到 dest

char strcat(char *dest,const char *src) 将字符串 src 添加到 dest 末尾

char strchr(const char *s,int c) 检索并返回字符 c 在字符串 s 中第一次出现的位置

int strcmp(const char *s1,const char *s2) 比较字符串 s1 与 s2 的大小,并返回 s1-s2

char strcpy(char *dest,const char *src) 将字符串 src 复制到 dest

size_t strcspn(const char *s1,const char *s2) 扫描 s1,返回在 s1 中有,在 s2 中也有的字符个数

char strdup(const char *s) 将字符串 s 复制到最近建立的单元

int stricmp(const char *s1,const char *s2) 比较字符串 s1 和 s2,并返回 s1-s2

size_t strlen(const char *s) 返回字符串 s 的长度

char strlwr(char *s) 将字符串 s 中的大写字母全部转换成小写字母,并返回转换后的字符串

char strncat(char *dest,const char *src,size_t maxlen) 将字符串 src 中最多 maxlen 个字符复制到字符串 dest 中

int strncmp(const char *s1,const char *s2,size_t maxlen) 比较字符串 s1 与 s2 中的前 maxlen 个字符

char strncpy(char *dest,const char *src,size_t maxlen) 复制 src 中的前 maxlen 个字符到 dest 中

int strnicmp(const char *s1,const char *s2,size_t maxlen) 比较字符串 s1 与 s2 中的前 maxlen 个字符

char strnset(char *s,int ch,size_t n) 将字符串 s 的前 n 个字符置于 ch 中

char strpbrk(const char *s1,const char *s2) 扫描字符串 s1,并返回在 s1 和 s2 中均有的字符个数

char strrchr(const char *s,int c) 扫描最后出现一个给定字符 c 的一个字符串 s

char strrev(char *s) 将字符串 s 中的字符全部颠倒顺序重新排列,并返回排列后的字符串

char strset(char *s,int ch) 将一个字符串 s 中的所有字符置于一个给定的字符 ch

size_t strspn(const char *s1,const char *s2) 扫描字符串 s1,并返回在 s1 和 s2 中均有的字符个数

char strstr(const char *s1,const char *s2) 扫描字符串 s2,并返回第一次出现 s1 的位置

char strtok(char *s1,const char *s2) 检索字符串 s1,该字符串 s1 是由字符串 s2 中定义的定界符所分隔

char strupr(char *s) 将字符串 s 中的小写字母全部转换成大写字母,并返回转换后的字符串