Input and output

level1

Professor JD

```
#include <stdio.h>
#include <math.h>
int main(){
float b,ls,rs1,rs2;
scanf("%f %f",&b,&ls);
b=b*b;
ls=ls*ls;
rs1=sqrt(ls-b);
rs2=sqrt(ls+b);
printf("%.5f %.5f",rs1,rs2);
        return 0;}
```

Lasya bought volleyball

```
#include <stdio.h>
int main()
{
float radiusofball,volumeofball;
scanf("%f",&radiusofball);
volumeofball=4/3*3.14*radiusofball*radiusofball*radiusofball;
printf("\n%f",volumeofball);
        return 0;
}
```

Sajid train ticket

```
#include <stdio.h>
int main()
{
int num1,num2,num3;
int sum;
scanf("%d %d %d",&num1,&num2,&num3);
sum=num1+num2+num3;
printf("\n%d",sum);
        return 0;
}
```

Brinta is playing chess

```
#include <stdio.h>
int main()
{
int n,m;
int d;
scanf("%d %d",&n,&m);d=n-1+(1+2*(n-1))*(m-1);
printf("%d",d);
        return 0;
}
```

Elavenil bakery

```
#include <stdio.h>
int main(){
int n;
scanf("%d",&n);
int f;
f=n/2+1;
printf("%d",f);
        return 0;}
```

One beautiful sunday

```
#include <stdio.h>
int main()
{
float num1,num2;
double resnum1,resnum2;
scanf("%f",&num1);
scanf("%f",&num2);
scanf("%lf",&resnum1);
scanf("%lf",&resnum2);
printf("\n%lf",num1);
printf("\n%lf",num2);
        return 0;
}
```

Tina brother friendly task

```
#include <stdio.h>
int main()
{int n;
int tot_square;
scanf("%d",&n);tot_square=(n*(n+1)/2)*(2*n+1)/3;
printf("%d",tot_square);
        return 0;
}
```

Electricity officer

```
#include <stdio.h>
#include <math.h>
int main()
{
int unitconsumed,totalbillamount;
int  costperunit;
scanf("%d",&unitconsumed);
scanf("%d",&costperunit);
totalbillamount=pow(unitconsumed,costperunit);
printf("\n%d",totalbillamount);
        return 0;
}
```

Rathik organised technic round

```
#include <stdio.h>
int main()
{
int testnum1,testnum2;
int sum,sub,mult,mod;
float div;
scanf("%d",&testnum1);
scanf("%d",&testnum2);
sum=testnum1+testnum2;
sub=testnum1-testnum2;
div=(float)testnum1/(float)testnum2;
mult=testnum1*testnum2;
mod=testnum1%testnum2;
printf("\nAddition : %d",sum);
printf("\nSubtraction : %d",sub);
printf("\nMultiplication : %d",mult);
printf("\nDivision : %0.3f",div);
printf("\nModulus : %d",mod);
        return 0;
}
```

Ramesh working eng clg

```
#include <stdio.h>
int main()
{
int alvqntoffood,messcnt,dividedqnt,remfood;
scanf("%d\t",&alvqntoffood);
scanf("%d",&messcnt);
dividedqnt=alvqntoffood/messcnt;
remfood=alvqntoffood%messcnt;
printf("%d %d\t",dividedqnt,remfood);
        return 0;
}
```

Binita was travelling from chennai to delhi

```
#include <stdio.h>
int main()
{
int tot_mins,hrs,mins;
scanf("%d",&tot_mins);
hrs=tot_mins/60;
mins=tot_mins%60;
printf("%d Hours and %d Minutes",hrs,mins);
            return 0;
}
```

Elavenil Chessboard

```
#include <stdio.h>
int main()
{int n,m;
int c;
scanf("%d %d",&n,&m);
c=(m-1)*(n-1);
printf("%d",c);}
```

Nancy Bought Apple

```
#include <stdio.h>
int main()
{
    int billamt,amtgiven;
    int q,r;
    scanf("%d",&amtgiven);
    scanf("%d",&billamt);
    q=amtgiven/billamt;
    r=amtgiven%billamt;
    printf("Quotient:%d\nRemainder:%d",q,r);

            return 0;
}
```

Tina’s Trainer

```
#include <stdio.h>
int main()
{  int U,V;
int sum;
scanf("%2d%2d",&U,&V);
sum=U*V/2+((U%2)*(V%2));
    printf("%d",sum);
            }
```

Arif Came from

```
#include <stdio.h>
int main()
{
    float val1,val2,outcome;
    scanf("%f%f",&val1,&val2);
    outcome=val1*val2;
    printf("%.4f",outcome);

	return 0;
}
```

Phoenix Mall

```
#include <stdio.h>
int main()
{
    int n,m,a;
    int stones;
    scanf("%d %d %d",&n,&m,&a);
stones=((n+a-1)/a)*((m+a-1)/a);
    printf("%d",stones);
	return 0;
}
```

Selvan was playing

```
#include <stdio.h>
int main()
{
    int length,width,height,surfacearea;
    scanf("%d %d %d",&length,&width,&height);
    surfacearea=2*(width*length+length*height+height*width);
    printf("%d", surfacearea);

	return 0;
}
```

IPL match

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

```
int main()
{
   int iplno;
   scanf("%d",&iplno);
   printf("%o",iplno);
   printf("\n%x",iplno);
        return 0;
}
```

Nathan works as HR

```
#include <stdio.h>
int main()
{
   float var1,var2,res;
   scanf("%f %f",&var1,&var2);
   res=var1+var2;
   printf("%.3f",res);

        return 0;
}
```

Employee of one million dollar

```
 #include <stdio.h>
int main()
{
   char Asc;
   scanf("%c",&Asc);
   printf("%d",(int)Asc);

        return 0;
}
```

Level 2

Issac loved to do agriculture

```
#include <stdio.h>
int main()
{
float tractLand,tractLandAcred;
scanf("%f",&tractLand);
tractLandAcred=(float)tractLand/43560;
printf("%.2f sq.ft is equal to %.2f acres",tractLand,tractLandAcred);
        return 0;
}
```

Rathik is young millionaire

```
#include <stdio.h>
int main()
{
float p,i,interest,amount;
int t;
scanf("%f %f %d",&p,&i,&t);
interest= p*i*t/100;
amount=p+interest;
printf("Interest after %d Years = $%.2f",t,interest);
printf("\nTotal Amount after %d Years = $%.2f",t,amount);
        return 0;
}
```

Salima saw a beautiful dress

```
#include <stdio.h>
int main()
{
int feet,inches;
float cms;
scanf("%d %d",&feet,&inches);
cms=feet*12*2.54+inches*2.54;
printf("Your height in centimeters is : %.2f",cms);
        return 0;
}
```

Aron took is gf binta

```
#include <stdio.h>
int main(){
int billwt;
float tax,tip,totaltax,totaltip,totalbill;
scanf("%d",&billwt);
tax=0.18;
tip=0.05 ;
totaltax=tax*billwt;
totaltip=tip*billwt;
totalbill=billwt+totaltax+totaltip;
printf("The Tax is %.2f",totaltax);
printf("\nThe Tip is %.2f",totaltip);
printf("\nTotal Bill With Tax and Tip is %.2f",totalbill);
        return 0;}
```

Roopa and athifa are sis

```
#include <stdio.h>
int main()
{
float num1,num2;
int sum;
scanf("%f",&num1);
scanf("%f",&num2);
sum=(int)num1+(int)num2;
printf("%d",sum);
        return 0;}
```

Sajid love super hero

```
#include <stdio.h>
#include <math.h>
int main()
{
int a,b,c;
float s,area;
scanf("%d %d %d",&a,&b,&c);
s=(a+b+c)/2;
area=sqrt(s*(s-a)*(s-b)*(s-c));
printf("%.2f\n",area);
        return 0;
}
```

Surya used to wear

```
#include <stdio.h>
int main(){
int sec,h,m,s;
scanf("%d",&sec);
h=sec/3600;
m=(sec-(h*3600))/60;
s=(sec-(h*3600)-m*60);
printf("%dH:",h);
printf("%dM:",m);
printf("%dS",s);
        return 0;}
```

Karthik working in HR

```
#include <stdio.h>
int main()
{
double salaryperday,totsalary;
int hour;
scanf("%d",&hour);
scanf("%lf",&salaryperday);
totsalary=(hour*salaryperday);
printf("%.2lf",totsalary);
        return 0;
}
```

Nathan was a student

```
#include <stdio.h>
int main()
{
int prodid,billid,quantity;
float price,totprice;
scanf("%d",&billid);
scanf("\n%d",&prodid);
scanf("\n%f",&price);
scanf("%d",&quantity);
totprice=price*quantity;
printf("%.2f",totprice);
return 0;
}
```

Arulmozivarmans dream come true

```
#include <stdio.h>
int main()
{
int GrossPayment,basic,da,hra;
scanf("%d %d %d",&basic,&da,&hra);
GrossPayment=(basic*(da+hra)/100)+basic;
double s=GrossPayment-0.5;
printf("%.lf",s);
        return 0;
}
```

Flipkart Announced

```
#include <stdio.h>
#include <math.h>
int main()
{
int N,fp,sp,tp;
scanf("%d",&N);
fp=pow(N,1);
sp=pow(N,2);
tp=pow(N,3);
printf("%d %d %d",fp,sp,tp);
        return 0;
}
```

Arul and Kani

```
#include <stdio.h>
int main()
{
float rad;
float PI=3.14,area,ci;
scanf("%f",&rad);
area=PI*rad*rad;
ci=2*PI*rad;
printf("%.2f\n%.2f",area,ci);
        return 0;
}
```

Nathan was a student

```
#include <stdio.h>
int main()
{
int prodid,billid,quantity;
float price,totprice;
scanf("%d",&prodid);
scanf("%d",&billid);
scanf("%f",&price);
scanf("%d",&quantity);
totprice=price*(float)quantity;
printf("%.2f",totprice);
        return 0;
}
```

Johnson was working

```
#include <stdio.h>
int main()
{
   int ndays,y,m,d;
   scanf("%d",&ndays);
   y= (int)ndays/365;
   ndays= ndays-(365*y);
   m= (int)ndays/30;
   d= (int)ndays-(m*30);
```

```
    printf("%d Y(s) %d M(s) %d D(s)", y, m, d);
          return 0;
}
```

2022 was approaching

```
#include <stdio.h>
int main()
{    int n,k; int x;
    scanf("%d %d",&n,&k);
    x=k/n;
    printf("%d",x);
          return 0;}
```

Ram was working

```
#include <stdio.h>
int main()binita
{
    int km; float x;
    float lpd;
    scanf("%d %f",&km,&lpd);
    x=km/lpd;
    printf("%.3f",x);

          return 0;
}
```

Athika and Ritu

```
#include <stdio.h>
int main()
{    float basic,sal;
    scanf("%f",&basic);
    sal=0.8*basic+0.4*basic+basic;
    printf("%.2f",sal);
          return 0;

}
```

Jannu and Preethi

```
#include <stdio.h>
int main()
{
    float base,height,area;
    scanf("%f %f",&height,&base);
    area=(height*base)/2;
    printf("%.3f",area);

          return 0;
}
```

Mallaiah has deposited

```
#include <stdio.h>
int main()
{
float amount,rate,time,si;
scanf("%f\n%f\n%f",&amount,&rate,&time);
si=(amount*rate*time)/100;
printf("%.4f",si);
          return 0;
}
```

Swathy and nancy

```
#include <stdio.h>
int main()
{
float spacenum;
scanf("%f",&spacenum);
int x=(int)spacenum;
printf("%d",x%10);
          return 0;
}
```

Arif planed to make a room

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

```
int main()
{
float length,width,area;
scanf("%f\n %f\n",&length,&width);
area=length*width;
printf("%.2f sq.ft",area);
        return 0;
}
```

Level 3

Darsh watch mechanic

```
#include <stdio.h>
int main()
{
int
days,hours,minutes,seconds,total_days_seconds,total_min_hours,total_minutes_
seconds,total;
scanf("%d",&days);
scanf("%d",&hours);
scanf("%d",&minutes);
scanf("%d",&seconds);
total_days_seconds=days*86400;
total_min_hours=hours*60;
total_minutes_seconds=(total_min_hours+minutes)*60;
total=total_days_seconds+total_minutes_seconds+seconds;
printf("%d seconds",total);
        return 0;
}
```

Nancy data scientist

```
#include <stdio.h>
int main()
{
int employeeID,areacode,hno,pincode;
scanf("%d",&hno);
scanf("%d",&pincode);
scanf("%d",&employeeID);
scanf("%d",&areacode);
printf("EmployeeID : %d\nArea Code : %d\nHouse Number : %d\nPincode :
%d",employeeID,areacode,hno,pincode);
return 0;
}
```

Zaher and vinod

```
#include <stdio.h>
int main()
{
float appleno;
scanf("%f",&appleno);
int t=(int)appleno;
 printf("%d",t%10);
          return 0;
}
```

Krishna arrive madura

```
#include <stdio.h>
#include <math.h>
int main(){
int m,n;
scanf("%d %d",&m,&n);
int no=ceil(m*n/(2.0*1));
printf("%d",no);
          return 0;}
```

Caleb physicist working DASA

```
#include <stdio.h>
#include <math.h>
int main()
{
float gravity,distance,vf;
gravity=9.8;
scanf("%f",&distance);
vf=sqrt(2*distance*gravity);
printf("%.2f m/s",vf);
          return 0;
}
```

Arav and nathan live in functional town

```
#include <stdio.h>
int main()
{
float a,b;
scanf("%f\n %f",&a,&b);
((a-b)<=0.5)?printf("Approximate number"):printf("Not an Approximate number");
          return 0;
}
```

Arulmozivarmans and is wife yazhini

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

```
int main()
{
int mpg;
float lph;
scanf("%d",&mpg);
lph=235.215/mpg;
printf("%.2f L/100 km",lph);
        return 0;}
```

Madhan worked as an local pilot

```
#include <stdio.h>
int main()
{
float distance,meter,feet,inches,centimeter;
scanf("%f",&distance);
meter=distance*1000;
feet=distance*3280.84;
inches=distance*39370.1;
centimeter=distance*100000;
printf("\n%.2f m",meter);
printf("\n%.2f ft",feet);
printf("\n%.2f in",inches);
printf("\n%.2f cm",centimeter);
        return 0;
}
```

Simon owned weld company

```
#include <stdio.h>
int main()
{
float celsius,fahrenheit;
scanf("%f",&celsius);
fahrenheit=(celsius*1.8)+32;
printf("%.2f fahrenheit",fahrenheit);
        return 0;
}
```

Vinod part of NGO

```
#include <stdio.h>
int main()
{
int year,yr;
scanf("%d",&year);
yr=year%100;
printf("%02d",yr);
        return 0;
}
```

Yasir was making a kite

```
#include <stdio.h>
#include <math.h>
int main()
{
    float s1,s2,s3,s,area;
    scanf("%f %f %f",&s1,&s2,&s3);
    s=(s1+s2+s3)/2;
    area=sqrt(s*(s-s1)*(s-s2)*(s-s3));
    printf("%.2f",area);
         return 0;
}
```

Ford once was

```
#include <stdio.h>
int main()
{    int seconds,days,hours,minutes;
    int s;
    scanf("%d",&s);
    days=s/(86400);
    s=s%86400;
    hours=s/3600;
    s=s%3600;
    minutes=s/60;
    seconds=s%60;
    printf("The Duration is %d days %d hours %d minutes %d
seconds",days,hours,minutes,seconds);
         return 0;
}
```

Satya is a mathematical

```
#include <stdio.h>
#include <math.h>
int main()
{
    double base,exp,opt;
    scanf("%lf %lf",&base,&exp);
    opt=pow(base,exp);
    printf("%.2lf",opt);
         return 0;
}
```

A pair of non-negative

```
#include <stdio.h>
int main()
{
   int k;
   long long n,ans;
   scanf("%d %lld",&k,&n);
   ans=((n/2)%1000000009);
   printf("%lld",(1+ans)%10000000009);

        return 0;
}
```

Shiva is a part of

```
#include <stdio.h>
int main()
{
   float base1,base2,height,area;
   scanf("%f %f %f",&base1,&base2,&height);
   area=0.5*(base1+base2)*height;
   printf("%.2f",area);
        return 0;
}
```

Swetha has N fruits

```
#include <stdio.h>
int main()
{int n;
scanf("%d",&n);
n%3==0?printf("YES"):printf("NO");
        return 0;
}
```

Binita always dreamed

```
#include <stdio.h>
int main()
{
int weight;
float height,bmi;
scanf("%d\n",&weight);
scanf("%f",&height);
bmi=(weight/(height*height));
printf("%.2f",bmi);
        return 0;
}
```

Nedumaran

```
#include <stdio.h>
#include<math.h>
int main()
{int price,loaves,regularprice;
float discountrate,discount,finalprice;
scanf("%d",&loaves);
price=185;
regularprice=loaves*price;
discount=0.6;
discountrate=(discount*regularprice);
finalprice=regularprice-discountrate;
printf("Regular Price=%d",regularprice);
printf("\nAmount Discounted=%.2f",discountrate);
printf("\nAmount to be paid=%.2f",finalprice);
        return 0;}
```

Tina successfully

```
#include <stdio.h>
int main()
{
float basicPay,employeeFund,employerFund;
scanf("%f",&basicPay);
employeeFund=(basicPay/100)*17.5;
employerFund=(basicPay/100)*23.5;
printf("%.2f\n%.2f",employeeFund,employerFund);
        return 0;
}
```

Arav was a popular

```
#include <stdio.h>
int main()
{
   int num,daop;
   scanf("%d",&num);
   daop=num%10;
   printf("%d",daop);

        return 0;
}
```

Flow control and operation

Level1

Vikram started programming

```
#include <stdio.h>
int main(){
int number1,number2;
scanf("%d %d",&number1,&number2);
if(number1<number2){
   printf("<");}
else if(number1>number2){
   printf(">");
}else{
   printf("=");
}
        return 0;}
```

Given an N integer

```
#include <stdio.h>
int main(){
int n;
float t;
scanf ("%d",&n);
if(n%2==0)
   printf("%d",(n/2)/n);
else{
   t=(n/2);
   t=(t+1)/n;
   printf("%.9f",t);}
        return 0;
}
```

Abi and janu

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

```
int n;
scanf("%d",&n);
if(n%8==1)
printf("%dLB",n+3);
else if(n%8==2)
printf("%dMB",n+3);
else if(n%8==3)
printf("%dUB",n+3);
else if(n%8==7)
printf("%dSU",n+1);
else if(n%8==0)
printf("%dSL",n-1);
else if(n%8==4)
printf("%dLB",n-3);
else if(n%8==5)
printf("%dMB",n-3);
else if(n%8==6)
printf("%dUB",n-3);
        return 0;
}
```

selvan work QC

```
#include <stdio.h>
int main()
{
char ch;
scanf("%c",&ch);
if((ch>=97&&ch<=122)||(ch>=65&&ch<=90)){
   printf("ALPHABET");
}
else{
   printf("NOT AN ALPHABET");
}
        return 0;
}
```

Primary maths

```
#include <stdio.h>
int main()
{int cp,sp,amt;
scanf("%d%d",&cp,&sp);
if(sp>cp)
{
   amt=sp-cp;
printf("Profit:%d",amt);
}
else if (cp>sp)
{
    amt=cp-sp;
    printf("Loss:%d",amt);
}
else
{
    printf("No Profit No Loss");
}
return 0;
}
```

Brittas parent buy puppy

```
#include <stdio.h>
int main()
{
int day;
scanf("%d",&day);
if(day==1)
    printf("Monday");
else if(day==2)
    printf("Tuesday");
else if(day==3)
    printf("Wednesday");
else if(day==4)
    printf("Thursday");
else if(day==5)
    printf("Friday");
else if(day==6)
    printf("Saturaday");
else if(day==7)
    printf("Sunday");
else
    printf("Invalid Input");
            return 0;
}
```

Pari is an architect

```
#include <stdio.h>
int main()
{
int l,b,area,peri;
scanf("%d",&l);
scanf("%d",&b);
area=l*b;
peri=2*(l+b);
if(area>peri){
   printf("Area");
   printf("\n%d",area);
}
else if(area<peri){
   printf("Peri");
   printf("\n%d",peri);
}
else{
   printf("Eq");
   printf("\n%d",peri);
}
         return 0;
}
```

In attack in war game

```
#include <stdio.h>
int main()
{
int a,b,c,d;
scanf("%d%d%d%d",&a,&b,&c,&d);
if(d>=b)
{
   printf("No");}
   else
   printf("Yes");
         return 0;}
```

Yasir is a techie work

```
#include <stdio.h>
int main()
{
int number;
scanf("%d",&number);
if(number<0){
   printf("NEGATIVE");
}
   else if(number>0){
      printf("POSITIVE");
   }
   else{
      printf("zero");
   }
         return 0;
}
```

Window of vinod room

```
}
```

```
#include <stdio.h>
int main()
{
int A,B,c;
scanf("%d %d",&A,&B);
c=A-B*2;
printf("%d",c);
        return 0;
}
```

Caleb and Irfan

```
#include <stdio.h>
int main()
{
   int apple1,apple2,apple3;
   scanf("%d %d %d",&apple1,&apple2,&apple3);
   if (apple2>apple1&&apple3>apple2) {
      printf("Fit into Budget");
   }
   else {
      printf("Dosen't fit into Budget");
   }
        return 0;
}
```

Aarav and Aaron

```
#include <stdio.h>
int main()
{    int aravspeed,aaronspeed,speeddiff;
   int speedmore;
   scanf("%d %d",&aravspeed,&aaronspeed);
   speeddiff=aravspeed-aaronspeed;
   speedmore=aaronspeed-aravspeed;
   if (speeddiff>speedmore){
      printf("%d",speeddiff);
   }
   else{
      printf("%d",speedmore); }
        return 0;
}
```

Tarun wants to print

```
#include <stdio.h>
int main()
{
    int N,n;    scanf("%d",&N);
    n=(N/2);
    if (N%2==0){
        printf("%d",n);
    }
    else{
        printf("%d",n+1);}
           return 0;}
```

Ram went to the bank

```
#include <stdio.h>
int main()
{
    int note500,note100,note50,note20,note10,note5,note2,note1;
    note500=note100=note50=note20=note10=note5=note2=note1=0;
    int amount;
    scanf("%d",&amount);
    if(amount>=500){
        note500=amount/500;
        amount-=note500*500;
    }
    if(amount>=100){
        note100=amount/100;
        amount-=note100*100;
    }
    if(amount>=50){
        note50=amount/50;
        amount-=note50*50;
    }
    if(amount>=20){
        note20=amount/20;
        amount-=note20*20;
    }
    if(amount>=10){
        note10=amount/10;
        amount-=note10*10;
    }
    if(amount>=5){
        note5=amount/5;
        amount-=note5*5;
    }
    if(amount>=2){
        note2=amount/2;
        amount-=note2*2;
    }
    if(amount>=1){
        note1=amount;
    }
    printf("500:%d\n",note500);
    printf("100:%d\n",note100);
    printf("50:%d\n",note50);
    printf("20:%d\n",note20);
    printf("10:%d\n",note10);
    printf("5:%d\n",note5);
    printf("2:%d\n",note2);
    printf("1:%d\n",note1);


           return 0;
}
```

Fazil frequently uses

```
#include <stdio.h>
int main()
{
    char X,Y;
    scanf("%c %c",&X,&Y);
    if (X>Y){
        printf(">");    }
    else if(X==Y){
        printf("=");    }
    else{
    printf("<");
    }
           return 0;}
```

Shivan is teaching

```
#include <stdio.h>
int main()
{
    int angle1,angle2,angle3,sumofangles;
    scanf("%d %d %d",&angle1,&angle2,&angle3);
    sumofangles=angle1+angle2+angle3;
    if(sumofangles==180){
        printf("Angles are valid");
    }
    else{
        printf("Angles are not valid");
    }

           return 0;
}
```

Election commission

```
#include <stdio.h>
int main()
{
    int age;
    scanf("%d",&age);
    if(age<18){
        printf("Not Eligible");
    }
    else{
    printf("Eligible");
    }
          return 0;
}
```

While Purchasing

```
#include <stdio.h>
int main()
{ int price,quantity,totexp;
    float d,c;
```

```
   scanf("%d %d",&quantity,&price);
   if(quantity>1000){
   c=price*0.1;d=price-c;
    totexp=(float)quantity*d;
    printf("%d",totexp);}
    else
    printf("%d",price*quantity);
    }
```

Rohit has A chocolate

```
#include <stdio.h>
int main()
{
int A,B,K;
scanf("%d %d %d",&A,&B,&K);
if(A>=K){
    printf("%d %d",A-K,B);
}
else if(A<=K)
{ printf("%d %d",0,B-(K-A));}
else {printf("%d %d",0,0);}
          return 0;
}
```

Three brothers

```
#include <stdio.h>
int main()
{int bro1,bro2,bro3,tallest;
scanf("%d%d%d",&bro1,&bro2,&bro3);
if(bro1>bro2 && bro1>bro3)
tallest=bro1;
else if (bro2>bro3)
tallest=bro2;
else
tallest=bro3;
printf("%d",tallest);
          return 0;}
```

Two Monkeys

```
#include <stdio.h>
int main()
{
int x1,x2,v1,v2;
scanf("%d %d %d %d",&x1,&x2,&v1,&v2);
if((x2-x1+v2-v1)%(v1-v2)!=0)
{
   printf("YES");
}
else
{
   printf("NO");
}
        return 0;
}
```

Tamil Selvan

```
#include <stdio.h>

int main()

{

 int note50,note20,note10,note5,note2,note1,amount;

 note50=note20=note10=note5=note2=note1=0;

 scanf("%d",&amount);

 if(amount>=50){

note50=amount/50;

 amount-=note50*50; }
 }

if(amount>=20){

 note20=amount/20;

amount-=note20*20;

 }

 if(amount>=10){

 note10=amount/10;

 amount-=note10*10;

 }

 if(amount>=5){

 note5=amount/5;

 amount-=note5*5;

 }

 if(amount>=2){

 note2=amount/2;

amount-=note2*2;

 }

 if(amount>=1){

 note1=amount;

 }
```

```
printf("50:%d\n",note50);

printf("20:%d\n",note20);

printf("10:%d\n",note10);

 printf("5:%d\n",note5);

printf("2:%d\n",note2);

printf("1:%d\n",note1);

 return 0;

}
```

Triple of Numbers

```
#include <stdio.h>
int main()
{
int a,b,c;
scanf("%d %d %d",&a,&b,&c);
if((a==c && b!=c)||(b==c && c!=a) ){
    printf("Yes");
}
else{       printf("No");     }
          return 0;
}
```

Laslya looking at a friend

```
#include <stdio.h>
int main()
{
    int year;
    scanf("%d",&year);
    if((year%4 == 0) || (year%400==0))
    {
        printf("LEAP YEAR");
    }
    else
    {
        printf("NOT A LEAP YEAR");
    }
          return 0;
}
```

Arulmozhivarman is working in ship

```
#include <stdio.h>
int main()
{   char ID;
    scanf("%c",&ID);

    if(ID == 'B'||ID=='b')
         printf("BattleShip");
    else if(ID=='C'||ID=='c')
         printf("Cruiser");
    else if(ID=='D'||ID=='d')
         printf("Destroyer");
    else if(ID=='F'||ID=='f')
         printf("Frigate");
   return 0;
}
```

Johit has two triangles

```
#include <stdio.h>
int main()
{   int a,b,c,d;
    scanf("%d %d %d %d",&a,&b,&c,&d); if((a*b)>=(c*d))
     printf("%d",a*b);
    else
     printf("%d",c*d);

   return 0;
}
```

Angean is a number of programming

```
#include <stdio.h>
int main()
{
int n,r,i;
scanf("%d %d",&n,&r);
if(10>=n) {
i=r+100*(10-n); printf("%d",i); }
else printf("%d",r);
        return 0; }
```

Agathiyan is the chief

```
#include <stdio.h>
int main()
{int N;
scanf("%d",&N);
if(N<10)
printf("Insufficient Earning");
else if(N<100)
printf("Very Low Earning");
else if(N<1000)
printf("Low Earning");
else if(N<10000)
printf("Sufficient Earning");
else if(N>10000)
printf("High Earning");

        return 0;
}
```

Swathi is working

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


{

   int angle1,angle2,angle3;
```

```
    scanf("%d %d %d",&angle1,&angle2,&angle3);

    if(angle1+angle2+angle3==180)
printf("Pizza Slice is Valid");
 else
printf("Pizza Slice is Not Valid");

          return 0;
}
```

Vishal is fighting

```
#include <stdio.h>
int main()
{int a,b;
scanf ("%d %d",&a, &b);
if(a%b==0)

printf("%d", a/b);
else if(a/b!=0)

printf("%d", (a/b)+1);
return 0;
}
```

Level2

Abilash and yazini

```
#include <stdio.h>
int main()
{
int month;
scanf("%d",&month);
switch(month){
    case 1:
    printf("31 days");
    break;
    case 2:
```

```
    printf("28/29 days");
    break;
    case 3 :
    printf("31 days");
    break;
    case 4:
    printf("30 days");
    break;
    case 5:
    printf("31 days");
    break;
    case 6:
    printf("30 days");
    break;
    case 7:
    printf("31 days");
    break;
    case 8:
    printf("31 days");
    break;
    case 9:
    printf("30 days");
    break;
    case 10:
    printf("31 days");
    break;
    case 11:
    printf("30 days");
    break;
    case 12:
    printf("31 days");
    break;
}
            return 0;}
```

Tina and fazil participate a contest

```
#include <stdio.h>
int main()
{
int n,a,b,k;
scanf("%d %d %d %d",&n,&a,&b,&k);
int count=0,f=0,i;
for(i=1;i<=n;i++){
if(i%a==0&&i%b!=0){
    count++;
    }
    else if(i%b==0){
        f++;
    }
}
    if(count+f>=k){
        printf("Win");
    }
else {
    printf("Lose");
}
            return 0;
}
```

Jackson work in restaurant

```
#include <stdio.h>
#include <math.h>
int main()
{
double n,v1,v2;
scanf("%lf %lf %lf",&n,&v1,&v2);
double t1,t2;
t1=1.414*n/v1;
t2=(2*n)/v2;
if(t1>t2){
    printf("Elevator");
}
else
    printf("Stairs");
            return 0;
}
```

Aarav new entrepreneur

```
#include <stdio.h>
int main()
{
int cp,sp;
scanf("%d",&cp);
scanf("%d",&sp);
if(cp>sp){
    printf("Loss");
}
else if(sp>cp){
    printf("Profit");
}
else{
   printf("No Profit No Loss");
}
         return 0;
}
```

Yesterday loki found k

```
#include <stdio.h>
int main()
{
int n,k;
scanf("%d %d",&n,&k);
if(n!=k){
   printf("NO");
}
else{ printf("YES");
}
         return 0;}
```

A team from royal squartaclub

```
#include <stdio.h>
int main()
{
int people_age,weight;
scanf("%d %d",&people_age,&weight);
if((people_age>=18)&&(weight>=40)){
   printf("Eligible for Donation");
}
else{
   printf("Not Eligible for Donation");
}
         return 0;}
```

Atifa withdraw

```
#include <stdio.h>
int main()
{
int amtreq;
float iniamt;
scanf("%d %f",&amtreq,&iniamt);
if(amtreq<iniamt){
float currentbalance=iniamt-amtreq-0.5;
printf("Current Balance : %.2f",currentbalance);
printf("\nInitial Balance : %.2f",iniamt);
}
else{
   printf("Invalid Withdrawal Request");
   printf("\nInitial Balance : %.2f",iniamt);
}
            return 0;
}
```

Mr.isaac head of tamilnadu

```
#include <stdio.h>
int main()
{
float celsius, fahrenheit;
scanf("%f",&fahrenheit);
celsius=(fahrenheit-32)*5/9;
if(celsius>=150){
    printf("%.2f Centigrade\nVery Hot",celsius);
}
else if(celsius>=100){
    printf("%.2f Centigrade\nHot",celsius);
}
else{
    printf("%.2f Centigrade\nModerate",celsius);
}
            return 0;
}
```

Paytm cashback

```
#include <stdio.h>
int main()
{
int currency;
scanf("%d",&currency);
(currency%2==0)?printf("Even Currency"):printf("Odd Currency");
            return 0;
}
```

Roy change profile

```
#include <stdio.h>
int main()
{
int l,w,h;
scanf("%d",&l);
scanf("%d %d",&w,&h);
if((w<l)||(h<l)){
    printf("UPLOAD ANOTHER");
}
else if(w==h){
    printf("ACCEPTED");
}
else{
    printf("CROP IT");
}
          return 0;
}
```

Aadi and Tara

```
#include <stdio.h>
int main()
{
    int month,numofdays;
    float roomrent,renttopay; float rentpay;
    scanf("%d %f %d",&month,&roomrent,&numofdays);
    renttopay=roomrent*numofdays;
    rentpay=renttopay+renttopay*0.2;
    if(month==4){
        printf("Rs.%.2f",rentpay);
    }
    else{
        printf("Rs.%.2f",renttopay);
    }
          return 0;
}
```

Mrs.Swathy

```
#include <stdio.h>
int main()
{
    int s1,s2,s3,s4,s5; float per;
    scanf("%d %d %d %d %d",&s1,&s2,&s3,&s4,&s5);
    per=(float)(s1+s2+s3+s4+s5)*100/500;
    printf("%.2f Percent",per);
    if(per>=90)
    printf("\nGrade A");
    else if(per>=80)
    printf("\nGrade B");
    else if(per>=70)
    printf("\nGrade C");
    else if(per>=60)
    printf("\nGrade D");
    else if(per>=40)
    printf("\nGrade E");
    else
    printf("\nGrade F");
           return 0;
}
```

I am not in danger

```
#include <stdio.h>
int main()
{
    int x,y;
    scanf("%d %d",&x,&y);int c;
    if((x-y)%2==0)
    c=(x>y)?1:3;
    else{if(x>y) c=2;
    else if (y>x) c=1;
    else c=0;}
     printf("%d",c);
           return 0;
}
```

Fazil and Yathra

```
#include <stdio.h>
int main()
{
    int a,b,c,d,n;
    scanf("%d %d %d",&a,&b,&n);
    c=a;
    d=b;
    for(int i=1;i<=n;i++)
    {
        if(i%2==1)
        c=c*2;
        else
        d=d*2;
    }
    if(c>=d)
    printf("%d",c/d);
    else
    printf("%d",d/c);
    return 0;
}
```

Karate demonstration

```
#include <stdio.h>
int main()
{
int s,w1,w2,w3;
scanf("%d %d %d %d",&s,&w1,&w2,&w3);
if(s>=w1+w2+w3){
   printf("1");
}
else if(s>=w1+w2){
   printf("2");
}
else if(s>=w2+w3){
   printf("2");
}
else{
   printf("3");
}
        return 0;
}
```

Elephant decided

```
#include <stdio.h>
int main()
{
int n,count=0;
scanf("%d",&n);
if(n%5==0){
   printf("%d",count=n/5);
}
else{
   printf("%d",count=n/5+1);
}
        return 0;
}
```

Shree and Harry

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

```
{float number1,number2,approx;
scanf("%f %f",&number1,&number2);
approx=number2-number1;
if(approx<=0.5)
printf("Approximate Number");
else
printf("Not an Approximate Number");
        return 0;
}
```

Caleb and Salima

```
#include <stdio.h>
int main()
{
int n1,n2,n3;
scanf("%d %d",&n1,&n2);
if(n1>n2)
{
n3=n1-n2;
printf("%d",n3);
}
else
printf("%d",n1+n2);
        return 0;
}
```

You are playing

```
#include <stdio.h>

int main()

{int n,k,x,y;

int x1,x2,x3,x4,y1,y2,y3,y4;

scanf("%d %d %d %d",&n,&k,&x,&y);

x1=x+n-x;

y1=y+n-x;

x2=y1;
y2=x1;

x3=x2-x2;

y3=y2-x2;

x4=y3;

y4=x3;

if(x1==y1)

printf("%d %d",x1,y1);

else

{    if(k%4==1)

    printf("%d %d",x1,y1);

    else if (k%4==2)

    printf("%d %d",x2,y2);

    else if (k%4==3)  printf("%d %d",x3,y3);   else    {printf("%d
%d",x4,y4);}}   return 0;}
```

Rashis classroom contains

```
#include <stdio.h>
int main()
{
    int n,m;
    scanf("%d%d", &n, &m);
    if(n%2!=0 && m%2!=0) printf("NO");
    else printf("YES");

            return 0;
}
```

level3

Yasir as N dairymilk

```
#include <stdio.h>
int main()
{
int n;
scanf("%d",&n);
if(n%3==0){
    printf("YES");
}
else{
    printf("NO");
}
        return 0;}
```

Sin and cos

```
#include <stdio.h>
int main()
{
long long int s,c,k,one=1,n;
scanf("%lld %lld %lld",&s,&c,&k);
n=s>=k?(one<<(s-k+1))|1:0;
if(k==1){
    if(s<=1)
    n+=c>0?(one<<(c+1))-2:0;
    else
    n+=c>=s?(one<<(c+1))-(one<<s):0;}
    else
    n+=s-k>=0&&s-k<c?one<<(s-k+1):0;
    printf("%lld",n);
        return 0;
}
```

Nathan two type of taxi

```
#include <stdio.h>
int main()
{
int D,Oc,Of,Od,Fs,Fb,Fm,Fd;
scanf("%d",&D);
scanf("%d %d %d",&Oc,&Of,&Od);
scanf("%d %d %d %d",&Fs,&Fb,&Fm,&Fd);
int olacost=Oc+(D-Of)*Od;
int ftcost=(D/Fs*60)*Fm+D*Fd+Fb;
if(olacost>ftcost){
    printf("Fastrack Taxi");
}
else if(ftcost>olacost){
   printf("OLA Taxi");
}
else{
   printf("OLA Taxi");
}
   return 0;
}
```

Maran head of data

```
#include <stdio.h>
int main()
{
int firstnum,secondnum;
scanf("%d %d",&firstnum,&secondnum);
printf("%d %d\n",firstnum--,++secondnum);
printf("%d %d\n",firstnum++,--secondnum);
printf("%d %d\n",firstnum--,++secondnum);
printf("%d %d\n",firstnum++,--secondnum);
printf("%d %d",firstnum,++secondnum);
         return 0;
}
```

Simon loves music

```
#include <stdio.h>
int main()
{
int L,D;
scanf("%d %d",&L,&D);
int sec=D/0.5;
int song=sec/L+1;
if(song!=sec){
printf("%d",song);
}
else{
   printf("%d",song);
}
         return 0;}
```

Arulmozivarans famous skill trainer

```
#include <stdio.h>
int main()
{
char operator;
double n1, n2;
scanf("%c",&operator);
scanf("%lf %lf",&n1,&n2);
switch(operator){
    case '+':
    printf("%.1lf",n1+n2);
    break;
    case '-':
    printf("%.1lf",n1-n2);
    break;
    case '*':
    printf("%.1lf",n1*n2);
    break;
    case '/':
    printf("%.1f",n1/n2);
    break;
}
            return 0;}
```

Simon work in casa

```
#include <stdio.h>
int main()
{
int side1,side2,side3;
scanf("%d %d %d",&side1,&side2,&side3);
if((side1==side2&&side2==side3)){
    printf("Equilateral triangle");
}
else if((side1==side2)||(side1==side3)||(side2==side3)){
    printf("Isosceles triangle");
}
else{
    printf("Scalene triangle");
}
            return 0;
}
```

Tina and fazil are bored

```
#include <stdio.h>
int main()
{
int x,y,k;
scanf("%d %d %d",&x,&y,&k);
int a=(x+y)/k;
if(a%2==0)
printf("Tina");
else
printf("Fazil");
        return 0;
}
```

You are given 2 points P and Q

```
#include <stdio.h>
#include <math.h>
int main()
{long long int px,py,pz,qx,qy,qz,dx,dy,dz,cx,cy,cz,r;
scanf("%lld%lld%lld%lld%lld%lld%lld%lld%lld%lld%lld%lld
%lld",&px,&py,&pz,&qx,&qy,&qz,&dx,&dy,&dz,&cx,&cy,&cz,&r);
double a=cx-px;
double b=cy-py;
double c=cz-pz;
double x=qx-px;
double y=qy-py;
double z=qz-pz;
double A=pow(b,2)+pow(c,2)-pow(r,2);
double B=pow(a,2)+pow(c,2)-pow(r,2);
double C=pow(b,2)+pow(a,2)-pow(r,2);
double E=dx*dx*A+dy*dy*B+dz*dz*C-2*b*c*dy*dz-2*a*c*dx*dz-2*a*b*dx*dy;
double F=2*(x*dx*A+y*dy*B+z*dz*C-b*c*y*dz-a*c*x*dz-b*c*z*dy-a*b*x*dy-
a*c*z*dx-a*b*y*dx);
double G=x*x*A+y*y*B+z*z*C-2*(b*c*y*z+a*c*x*z+a*b*x*y);
double qw=sqrt(F*F-4*E*G);
double ans;
if(E)
ans=(qw-F)/(2*E);
else
ans=(-1*G)/F;
printf("%.10lf\n",ans);
return 0;
}
```

Yasir is chief in charge

```
#include <stdio.h>
int main(){
char gender;
scanf("%c",&gender);
switch(gender){
   case 'M':
   printf("Male");
   break;
   case 'm':
   printf("Male");
   break;
```

```
    case 'F':
    printf("Female");
    break;
    case 'f':
    printf("Female");
    break;
    default:
    printf("Unspecified Gender");
}
            return 0;}
```

Today is Darsh Birthday

```
#include <stdio.h>
int main()
{
    int favorite_number,first_number,difference;
    scanf("%d %d %d",&first_number,&favorite_number,&difference);
    if((first_number-favorite_number)%difference==0){
        printf("YES");
    }
    else{
        printf("NO");
    }

            return 0;
}
```

Central Library

```
 #include <stdio.h>
#include <stdlib.h>
#include <assert.h>
#include <math.h>
int main()
{
    int d1,d2,m1,m2,y1,y2;
    scanf("%d %d %d",&d1,&m1,&y1);
    scanf("%d %d %d",&d2,&m2,&y2);

     if((d1>d2)&&(m1==m2)&&(y1==y2))
    printf("%d",(15*(d1-d2)));
    else if((m1>m2)&&(y1<=y2))
    printf("%d",(500*(m1-m2)));
    else if((d1<d2)&&(m1<m2)&&(y1>y2))
    printf("10000");
    else
    printf("10000");
    return 0;
}
```

Salima is working

```
#include <stdio.h>
#include <math.h>
int main()
{
    float a,b,c; float root1,root2,imaginary; float discriminent;
    scanf("%f %f %f",&a,&b,&c);
    discriminent=(b*b)-(4*a*c);
    switch(discriminent>0)
    {
    case 1:
    root1=(-b+sqrt(discriminent))/(2*a);
    root2=(-b-sqrt(discriminent))/(2*a);
    printf("%.2f %.2f",root1,root1);
    break;
    case 0:
    switch(discriminent<0)
    {
            case 1:
            root1=root2=-b/(2*a);
            imaginary=sqrt(-discriminent)/(2*a);
            printf("%.2f + i%.2f and %.2f - i
%.2f",root1,imaginary,root2,imaginary);
            break;
            case 0:
            root1=root2=-b/(2*a);
            printf("%.2f %.2f",root1,root2);
            break;
    }
   }
   return 0;
}
```

Simon, nancy and Yasir

```
#include <stdio.h>
int main()
{
    char alphabet;
    scanf("%c",&alphabet);
    switch(alphabet)
    {
        case 'a':
        printf("Vowel"); break;
        case 'e':
        printf("Vowel"); break;
        case 'i':
        printf("Vowel"); break;
        case 'o':
        printf("Vowel"); break;
        case 'u':
        printf("Vowel"); break;
        case 'A':
        printf("Vowel"); break;
        case 'E':
        printf("Vowel"); break;
        case 'I':
        printf("Vowel"); break ;
        case 'O':
        printf("Vowel"); break;
        case 'U':
        printf("Vowel"); break;
        default:
        printf("Consonant"); break;
    }

           return 0;
}
```

Nathan is so fashion

```
#include <stdio.h>
int main()
{
    int days;
    scanf("%d",&days);
    switch(days)
    {
        case 1:
        printf("Azure"); break;
        case 2:
        printf("Beige"); break;
        case 3:
        printf("Brick Red"); break;
        case 4:
        printf("Champagne"); break;
        case 5:
        printf("Desert sand"); break;
        case 6:
        printf("Ivory"); break;
        case 7:
        printf("Pear"); break;
        default:
        printf("Invalid Day"); break;

    }


          return 0;
    }
```

Nancy is a graduate

```
      #include <stdio.h>
int main()
{int travelmode;
scanf("%d",&travelmode);
switch(travelmode){
    case 1: printf("Car is booked");
                    break;
    case 2: printf("Bus is booked");
                    break;
    case 3: printf("Flight is booked");
                    break;
    default :printf("Invalid Request");
                    break;
}
```

```
          return 0;
}
```

Lee is ill

```
 #include <stdio.h>
int main()
{
int lengthofbook,numofpages;
scanf("%d %d",&lengthofbook,&numofpages);
if(lengthofbook<=23&&numofpages<=1000&&numofpages>=500){
    printf("Take Medicine");
}
else{
    printf("Don't take Medicine");}
          return 0;}
```

Selvan is one of the highest

```
#include <stdio.h>
int main()
{
int workalloid;
scanf("%d",&workalloid);
switch(workalloid){
    case 101: printf("Cinematographer");
            break;
    case 201: printf("Editor");
            break;
    case 301: printf("Marketing Manager");
            break;
    case 401: printf("Content Engineer");
            break;
    case 501: printf("Editorial Assistant");
            break;
}

          return 0;}
```

Nowadays many

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

```
scanf("%d",&workage);
if(workage<18){
    printf("You are Minor\n");
    printf("Continue Your Studies");
}
else if(workage>=18&&workage<=60){
printf("You are Eligible\n");
printf("You can Apply for Job");}
else{
    printf("You are too Old\n");
    printf("Pls Collect your Pension");
}
           return 0;
}

  #include <stdio.h>
int fair(int a1,int a2,int c1,int c2)
{
    if((a1>a2 && c1>c2) || (a1<a2 && c1<c2) || (a1==a2 && c1==c2))
    {
        return 1;
    }
    else
    {
        return 0;
    }
```

Pongal gift

```
#include <stdio.h>
int fair(int a1,int a2,int c1,int c2)
{
    if((a1>a2 && c1>c2) || (a1<a2 && c1<c2) || (a1==a2 && c1==c2))
    {
        return 1;
    }
    else
    {
        return 0;
    }
}
int main()
{
```

```
int a1,a2,a3,c1,c2,c3;
scanf("%d %d %d %d %d %d",&a1,&a2,&a3,&c1,&c2,&c3);
if(fair(a1,a2,c1,c2) && fair(a1,a3,c1,c3) && fair(a3,a2,c3,c2))
{
    printf("FAIR");
}
else if(6<5)
{
    a1=a1+a2;
}
else
{
    printf("NOT FAIR");
}
}
```

Array n loop

level1

After complete serious investigation

```
#include <stdio.h>
int main()
{
    int t,n,h,i,l=1,count;
    scanf("%d",&t);
    while(t--)
    {
        l=1;
        count=0;
        scanf("%d",&n);
        for(i=1;i<=n;i++)
        {
            scanf("%d",&h);
            if(h==l)
            {
                count+=2;
            }
            if(h>l)
            {
            l=h;
            count++;
            }
        }
        printf("%d\n",count);
    }
            return 0;
}
```

Today is caleb birthday

```
#include <stdio.h>
int main()
{int t,n,m,k;
scanf("%d", &t);
while(t--){
    scanf("%d %d %d",&n, &m, &k);
    if((m>n && k>= (m-n)) || (n>m && k>= (n-m))){
        printf("0\n");
    }
    else if(n>m && k<(n-m)){
        printf("%d\n",n-(m+k));
    }
    else{printf("%d",m-(n+k));}
}
            return 0;
}
```

Memory n crow

```
#include <stdio.h>
int main()
{
int competition[100002];
int n;
scanf("%d",&n);
int i,sum;
for(i=0;i<n;i++)
    scanf("%d",&competition[i]);
    for(i=0;i<n;i++){
        sum=competition[i]+competition[i+1];
        printf("%d ",sum);
    }
            return 0;
}
```

Caleb challenge selvan

```
#include <stdio.h>
#include<math.h>
void Clac_square(long int start,long int end){
    int i,count=0;
    for(i=start;i<=end;i++){
        int n=sqrt(i);
        if(n==sqrt(i))
        count++;}
        printf("%d\n",count);
}
int main(){
    long int q,start,end;
    scanf("%ld",&q);
    while(q--){
        scanf("%ld %ld",&start,&end);
        Clac_square(start,end);
}
            return 0;
}
```

Akash fan of A.R.Rahman

```
#include <stdio.h>
int main()
{
int nooffamilymembers;
scanf("%d",&nooffamilymembers);
for(int i=nooffamilymembers;i>=1;i--){
for(int j=i;j>=1;j--){
    printf("%d ",i);
}
printf("\n");
}
            return 0;
}
```

Teddy bear

```
#include <stdio.h>
int main()
{
    int x[100],y[100],u[100],v[100];
    int i,k,n;
    scanf("%d %d",&n,&k);
    for(i=0;i<k;i++){
        scanf("%d",&x[i]);
    }
    for(i=0;i<k;i++)
```

```
        scanf("%d",&y[i]);
    u[0]=x[0];
    v[0]=y[0];
    if(u[0]==2 && v[0]==1)
        printf("3");
    else if(u[0]==2)
        printf("1");
    else if(u[0]==0)
        printf("4");
    else
        printf("6");
            return 0;
}
```

Teddy bear (updated ans)

```
#include <stdio.h>

int x[100],y[100],u[100],v[100];
int main()  {
    int k, n, m; long long a = 0;
    int i;
    scanf("%d %d",&n,&k);
    m = n-1;
    for ( i = 0; i < k; ++i)
        scanf("%d",x+i), --x[i], u[i] = m-x[i];
    for ( i = 0; i < k; ++i)
        scanf("%d",y+i), --y[i], v[i] = m-y[i];
    for ( i = 0; i < k; ++i)
        if (x[i] < u[i])
            a += x[i];
        else
            a += u[i];
    for ( i = 0; i < k; ++i)
        if (y[i] < v[i])
            a += y[i];
        else
            a += v[i];
    printf("%lld",a);
    return 0;
}
```

Matriculation school

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

```
int rows;
scanf("%d",&rows);
for(int i=1;i<=rows;i++){
    for(int j=1;j<=i;j++){
        if(i==1||i==rows||j==1||j==i){
            printf("1 ");
        }
        else{
            printf("0 ");
        }
    }
    printf("\n");
}
            return 0;
}
```

PUBG

```
#include <stdio.h>
int main()
{int i,j,row,col,t,moves;
long long int g,grid[100][50],coins[50];
for(i=0;i<100;i++){
    grid[i][0]=1;
    for(j=1;j<=i&&j<50;j++){
        if(i==j)
        grid[i][j]=1;
        else
        grid[i][j]=grid[i-1][j-1]+grid[i-1][j];
    }
}
scanf("%d",&t);
while(t--){
    scanf("%d %d %lld",&row,&col,&g);
    moves=0;
    while(g>0){
        row=col;
        while(row<100&&grid[row][col]<=g)
        row++;
        row--;
        g=g-grid[row][col];
        coins[moves]=grid[row][col];
        moves++;
        col--;
    }
    printf("%d\n",moves);
    for(i=0;i<moves;i++)
    printf("%lld ",coins[i]);
    printf("\n");
}
            return 0;
}
```

hassan and roopa

```
#include <stdio.h>
int main()
{
int t,n,x[100002],y[100002];
scanf("%d",&t);scanf("%d",&n);
int i,sum1=0,sum2=0;
while(t--){
    for(i=0; i<n;i++){
        scanf("%d",&x[i]);}
        for(i=0;i<n;i++){
            scanf("%d",&y[i]);
        }
        for(i=0;i<n;i++){
            if(i%2==0){
                sum1+=x[i];
                sum2+=y[i];
            }
            else{
                sum1+=y[i];
                sum2+=x[i];
            }
        }
        (sum1<sum2)?printf("%d",sum1):printf("%d",sum2);
    }
            return 0;
}
```

Lasya with friends

```
#include <stdio.h>
int main()
{
    int n;
int i,j,k;
scanf("%d",&n);
for(i=1;i<=n;i++){
       if(i%2==0){
           k=2;
       }
    else{
      k=1;
    }
    for(j=1;j<=i;j++){
      printf("%d ",k);
      k +=2;
    }
    printf("\n");
}
       return 0;}
```

Mr. Arulmozhivalman loves programming

```
#include <stdio.h>
int main()
{
    int Size;
    int i,j,count=0;
    int FreqArr[100000];
    scanf("%d",&Size);
    for(i=0;i<Size;i++)
    {
        scanf("%d",&FreqArr[i]);
    }
    for(i=0;i<Size;i++)
    {
        for(j=i+1;j<Size;j++)
        {
            if(FreqArr[i]==FreqArr[j])
            {
                count ++;
                printf("%d",FreqArr[i]);
                return 0;
            }
        }
    }

}
```

Steve Waugh and Mark Waugh

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

```
int i;
int main()
{   int markwaugh,stevewaugh,n;
    scanf("%d",&n);
    int arr[n+1];
    arr[0] = 1;
       arr[1] = 1;
       arr[2] = 2;
       for (i = 3; i <=n; i++)
       arr[i] = arr[i - 1] + arr[i - 2]+ arr[i - 3];
     stevewaugh=arr[n];
   for(i=2;i<=n;i++)
    arr[i]=arr[i - 1] + arr[i - 2];
    markwaugh=arr[n];
   printf("Steve Waugh:%d\nMark Waugh:%d",stevewaugh,markwaugh);
    return 0;
}
```

Little Lion King

```
#include <stdio.h>
int main(void)
{    int T,N,C; int a;
    scanf("%d",&T);
    while(T--){
        int i;
        scanf("%d %d",&N,&C);
        for(i=0;i<N;i++){
            scanf("%d",&a);
            C-=a;       }

        if(C>=0)
           printf("Yes\n");
         else
           printf("No\n");    }
           return 0;}
```

Let's consider a triangle

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

```
{
    int t,n,i,j;
    scanf("%d",&t);
    while(t--)
    {
        scanf("%3d",&n);
        int ar[n][n];
        for(i=0;i<n;i++)
    {
        for(j=0;j<=i;j++)
        scanf("%3d",&ar[i][j]);
    }
    for(i=n-1;i>=0;i--)
    {
        for(j=0;j<i;j++)
    {
        if(ar[i][j]>ar[i][j+1])
        ar[i-1][j]+=ar[i][j];
        else
        ar[i-1][j]+=ar[i][j+1];
    }
    }
    printf("%d\n",ar[0][0]);
    }

           return 0;
}
```

Advika bought cadbury

```
#include <stdio.h>
int main()
{
    int r,c;
    int arr[100][100],sum = 0,i,j;
    scanf("%d %d",&r,&c);
    for(i = 0;i < r;i++){
        for(j=0;j < c;j++){
            scanf("%d",&arr[i][j]);
        }
    }
    for(i=0;i<r;i++){
        for(j=0;j<c;j++){
            if(i==0 || j==0 || i==r-1 || j==c-1){
                sum+=arr[i][j];
            }
        }
    }
    printf("%d",sum);
           return 0;
}
```

Rakesh given an array

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int j=0,t,i,n,k;
    scanf("%d",&t);
    while(j<t)
    {
        scanf("%d %d",&n,&k);
        int integers[n];
        int flag=0;
        for(i=0;i<n;i++)
        scanf("%d",&integers[i]);
        for(i=0;i<n;i++)
        if(abs(integers[i]-integers[i+1])<=k)
        flag++;
        if(flag==(n-1))
        printf("\nYES");
        else
        printf("\nNO");
        j++;
    }

           return 0;
}
```

Hero of the Story

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

```
bool check(int arr[],int n,int m,long long mid)
{
    int days=0,i;
    long long temp=0;
    for(i=0;i<n;i++)
    {if(temp+arr[i]>mid){
            temp=0;
            days++;
            temp+=arr[i];}
        else{
            temp+=arr[i];}
    }
    return days<=m-1&&temp<=mid;
}
int main()
{
    int n,m,a,mintime=0;
    scanf("%d %d",&n,&m);
    int timetosolve[100005];
    for(a=0;a<n;a++){
        scanf("%d",&timetosolve[a]);
        if(timetosolve[a]>mintime){
            mintime=timetosolve[a];
        }
    }
    long long lo=mintime,hi=1e10,mid;
    while(hi-lo>1)
    {
        mid=(hi+lo)/2;
        if(check(timetosolve,n,m,mid))
{
    hi=mid;
}
else
{
    lo=mid+1;
}}
lo=check(timetosolve,n,m,lo)?lo:hi;
printf("%lld\n",lo);

           return 0;}
```

Arif has N lights

```
#include <stdio.h>
int main()
{
int lights[100001],n,q,i,a,b;
scanf("%d %d",&n,&q);
for(i=0;i<n;i++)
{scanf("%d",&lights[i]);}
while(q-->0)
{scanf("%d %d",&a,&b);
for(i=a-1;i<b;i++)
lights[i]=!lights[i];
}
for(i=0;i<n;i++){
    printf("%d ",lights[i]);}
           return 0;
}
```

Ganapathy mathematician

```
#include <stdio.h>
int main()
{int noofrowsinclass;
int i,j;
scanf("%d",&noofrowsinclass);
for(i=1;i<=noofrowsinclass;i++){
for(j=1;j<=i;j++){
    if(i%2==0)
        printf("Fail ");
else
printf("Pass ");}
printf("\n");
}

           return 0;
}
```

Nathan is a researcher

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

```
{int arr1[26],arr2[26];
char str[10001];
int t,i,index;
scanf("%d",&t);
while(t--){
    scanf("%s",str);
    int len = strlen(str);
    memset(arr1,0,sizeof(arr1));
    memset(arr2,0,sizeof(arr2));
    for(i = 0; i < len/2; i++){
        index = str[i] - 'a';
        arr1[index]++;
    }
    for(i = (len +1)/2; i < len; i++){
        index = str[i] - 'a';
        arr2[index]++;
    }
    int test = 0;
    for(i=0; i < 26; i++)
    if(arr1[i] !=arr2[i])
    test = 1;
    (test ==0) ? printf("YES\n") : printf("NO\n");
}

           return 0;
}
```

You like tracking

```
#include <stdio.h>
int main()
{
int h[100001];
int i,j,max=0,n;
scanf("%d",&n);
for(i=0;i<n;i++)
{scanf("%d",&h[i]);
if(h[i]>max)
max=h[i];
}
int s[100]={0};
for(i=0;i<n-1;i++)
{if(h[i]>h[i+1])
{
    for(j=h[i+1];j<h[i];j++)
    s[j]++;
}
if(h[i]<h[i+1])
{
    for(j=h[i];j<h[i+1];j++)
    s[j]++;
}
}
int k=0;
for(j=0;j<max;j++)
{if(s[j]>k)
k=s[j];
}
printf("%d",k);
          return 0;
}
```

Software Tool

```
#include <stdio.h>
int main()
{int t,i;
scanf("%d",&t);
while(t--){
    int n,m,k,c1=0,c2=0;
    scanf("%d%d%d",&n,&m,&k);
    int a[m],b[k],c[101];
    for(i=0;i<=101;i++)
    c[i]=0;
    for(i=0;i<m;i++){
        scanf("%d",&a[i]);
        c[a[i]]++;
    }
    for(i=0;i<k;i++){
        scanf("%d",&b[i]);
        c[b[i]]++;
    }
    for(i=0;i<101;i++){
        if(c[i]==2){
            c1++;}
```

```
      }
      for(i=1;i<=n;i++){
         if(c[i]==0){
            c2++;
   }}
      printf("%d %d\n",c1,c2);
 }
        return 0;
}
```

Rainbow

```
#include <stdio.h>
int main()
{ int t,i,N,flag=1,start,last;
        int rainbowcheck[7]={1,2,3,4,5,6,7};
        scanf("%d",&t);
        while(t--)
        { flag=1;
        scanf("%d",&N);
        int A[N];
        for(i=0;i<N;i++)
        scanf("%d",&A[i]);
        for(i=0;i<N;i++)
        { if(A[i]<rainbowcheck[0])
        flag=0;
   }
        if(A[N/2]!=7)
        flag=0;
        start=0;
        last=N-1;
        while(start<last)
        { if(A[start]!=A[start+1]&&A[start+1]-A[start]!=1)
        flag=0;
        if(A[start]!=A[last])
        flag=0;
        start++;
        last--;
   }
        if(flag)
        printf("yes\n");
        else
```

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

Fazil loves to perform

```
#include <stdio.h>
int main()
{
    int rot[100000];int n,k,j,i,t,tc;
    scanf("%d",&tc);
    while(tc--)
    {
        scanf("%d%d",&n,&k);
        for(j=0;j<n;j++)
            scanf("%d ",&rot[j]);
        for(j=0;j<k;j++)
            {
                t=rot[n-1];
                for(i=n-1;i>0;i--)
                    rot[i]=rot[i-1];

                rot[i]=t;
            }
            for(i=0;i<n;i++)
                printf("%d ",rot[i]);

            printf("\n");
    }

           return 0;
}
```

Mahesh has given 2 dimensional

```
#include <stdio.h>
int main()
{
    int A[3][3];
    int i,j;
    for(i=0;i<3;i++){
        for(j=0;j<3;j++){
            scanf("%d ",&A[i][j]);
        }
    }
    int s1=0,s2=0;
    for(i=0;i<3;i++){
        for(j=0;j<3;j++)
        {
            if((i+j)%2==0)
                s1=s1+A[i][j];
        else
            s2=s2+A[i][j];
        }
    }
    printf("%d\n%d",s1,s2);
return 0;
}
```

Once N men and M women

```
0#include <stdio.h>
#include <string.h>
int main()
{
int t,men,women,collisions=0;
scanf("%d",&t);
scanf("%d %d\n",&men,&women);
int i,sum=0;
char a[men][women];
for(i=0;i<men;i++)
    for(t=0;t<women;t++)
    scanf("%c ",&a[i][t]);
for(i=0;i<women;i++)
{
    for(t=0;t<men;t++)
    if(a[t][i]=='1')
    collisions+=1;
    sum+=collisions*(collisions-1)/2;
    collisions=0;
    }
printf("%d",sum);
           return 0; }
```

Joslyn like problems

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

```
int main()
{
int T;
int i,j;
scanf("%d",&T);
while(T--)
{
    int count=0;
    int matprob[100];
    int n;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        scanf("%d",&matprob[i]);
    }
    for(i=0;i<n;i++)
    {
        int sum=0,product=1;
        for(j=i;j<n;j++)
        {
            sum+=matprob[j];
            product*=matprob[j];
            if(sum==product)
            count++;
        }
    }
    printf("%d\n",count);
}
           return 0;
}
```

Mukesh and salima

```
#include <stdio.h>
int main(){
    int numofapples, x, i, c=0;
    scanf("%d", &numofapples);
    int arr[100]={0};
    for(i=0; i<numofapples; ++i){
        scanf("%d", &x);
        arr[x]++;
    }
    for(i=0; i<100; ++i){
        if(arr[i]>1) c += (arr[i]-1);
    }
    printf("%d", c);
           return 0;
}
```

Yogesh booked the ticket

```
#include <stdio.h>
int main(){
    int noofrows,i,j;
    scanf("%d",&noofrows);
    for(i=0;i<noofrows;i++){
        for(j=0;j<=i;j++){
            printf("%d ",i+1);
        }
        printf("\n");
    }



           return 0;
}
```

Nathan is a researcher

```
#include <stdio.h>
#include <string.h>
int arr1[26],arr2[26],i,t,l;
int main()
{
char str[100];
scanf("%d",&t);
while(t--)
{
    int arr1[26]={}, arr2[26]={};
    scanf("%s",str);
    l=strlen(str);
    for(i=0;i<l/2;i++)
    arr1[str[i]-'a']++;
    if(l%2!=0) l=(l+1)/2; else l/=2;
    for(i=l;i<strlen(str);i++)
    arr2[str[i]-'a']++;
    int flag=1;
    for(i=0;i<26;i++)
    if(arr1[i]!=arr2[i]) flag=0;
    (flag==1)?printf("YES\n"):printf("NO\n");
}
               return 0;
}
```

Ambikapathy wants to

```
#include<stdio.h>
#include<stdbool.h>
int main()
    {
        int k,m,i=0,j=0;
        scanf("%d %d",&k,&m);
        int lights[m];
        for(i=1;i<=m;i++)
            {
                lights[i]=0;
            }
        while(k--)
            {
                int X;
                scanf("%d",&X);
                int arr2[X];
                for(i=1;i<=X;i++)
                    { scanf("%d",&arr2[i]);       }
                for(i=1;i<=m;i++)
                    {
                        for(j=1;j<=X;j++)
                            {
                                if(arr2[j]==i)
                                    {
                                        lights[i]++;
                                    }
                            }             }
                     }
            bool flag=true;
            for(i=1;i<=m;i++)
                {
                    if(lights[i]==0)
                        {
                            flag=false;
            break;
          }
        else
          {  flag=true;   }
      }
    if(flag==true)
      {
        printf("YES\n");
      }
    else if(flag==false)
      {
        printf("NO\n");
      }
return 0;   }
```

Level2

Yasir has an array aops

```
#include <stdio.h>
int main()
{int n,q,aops[100000];
int i,t[4];
scanf("%d %d",&n,&q);
for(i=0;i<n;i++){
  scanf("%d\n",&aops[i]);
}
for(i=0;i<q;i++){
  scanf("%d\n",&t[i]);
  if((t[i]<=aops[0]&&t[i]>=aops[1])||(t[i]>=aops[0]&&t[i]<=aops[1]))
  printf("Yes\n");
  else
  printf("No\n");
}
      return 0;
}
```

You probably know

```
#include <stdio.h>
int main()
{int t,n,m,i;
scanf("%d\n%d %d",&t,&n,&m);
int a[m],set=0;
while(t--){
    for(i=0;i<m;i++){scanf("%d",&a[i]);}
    if(m%2==0)
    set=1;
    else
    set=2;
}
printf("%d",set);
            return 0;
}
```

Yasir has array of positive integers

```
#include <stdio.h>
int main()
{
    int i,t,a[100000],b[100000],j=0;
    scanf("%d",&t);
    while(t--){
        int n;
        scanf("%d",&n);
        for(i=0;i<n;i++)scanf("%d",&a[i]);b[i]=0;
        b[j++]=a[n-1];
        for(i=n-1; i>=0;i--)if( a[i] >= b[j-1]){b[j] = a[i];j++;}
            for(j=j-1; j>0;j--) printf("%d ",b[j]);
          printf("\n");
    }
            return 0;
}
```

Brita and swaty

```
#include <stdio.h>
int main()
{int t,a,n;
scanf("%d",&t);
int p[2];
while(t--){
    int i;
for(i=0;i<2;i++){
    scanf("%d",&p[i]);}
    n=p[0];a=p[1];
    if((a%2==0&&n%2==0)||(n<a&&n%2==0)){printf("Swathy\n");}
    else{printf("Britta\n");}
}
            return 0;
}
```

There are N students

```
#include <stdio.h>
int main()
{int t,n,a[10002],b[10002];
int i, count=0;
scanf("%d",&t);
while(t!=0){
    scanf("%d",&n);
    for(i=0;i<n;i++)
    scanf("%d",&a[i]);
    for(i=0;i<n;i++)
    scanf("%d",&b[i]);
    for(i=0;i<n;i++){
        if((a[i]-a[i-1])>=b[i])
        count++;
    }
    printf("%d\n",count);
    count=0;t--;
}
            return 0;
}
```

Once diya sale tv

```
#include <stdio.h>
#include <stdlib.h>
int cmpfunc(const void *a, const void *b){
    return(*(int*)a - *(int*)b);}
int main()
{int n,m,price[104];
int s=0;
int i;
scanf("%d %d",&n,&m);
for(i = 0; i < n; i++){
    scanf("%d",&price[i]);}
qsort(price,n,sizeof(int), cmpfunc);
int x=0;
while(m>0){
    if(price[x]>0){m=0;}
    else{s = s + price[x];x++;
    m--; }
}
printf("%d",-s);
return 0;
}
```

Arulmozivaran invited N friends

```
#include <stdio.h>
#define N 1000
int main()
{int t;
scanf("%d",&t);
while(t-->0){
    static int adj[N][N],qq[N],aa[N];
    int n,m,h,i,j,no;
    int k,q;
    scanf("%d %d",&n,&m);
    for(i=0;i<n;i++)
    for(j=0;j<n;j++)
    adj[i][j]=0;
    for(h=0;h<m;h++){
        scanf("%d %d",&i,&j);
        i--,j--;
        adj[i][j]=adj[j][i]=1;
    }
    for(i=0;i<n;i++)
    aa[i]=-1;
    no=0;
    for(i=0;i<n;i++)
    if(aa[i]==-1){
        k=q=0;
        aa[i]=0,qq[k+q++]=i;
        while(q>0){
            int i=qq[k++];
            q--;
            for(j=0;j<n;j++)
            if(i!=j&&!adj[i][j]){\
            if(aa[j]==-1)
            aa[j]=aa[i]^1,qq[k+q++]=j;
            else if(aa[i]==aa[j])
            no=1;
        }
    }
}
printf("%s\n",no==0?"YES":"NO");
}
            return 0;
}
```

snowbell

```
#include <stdio.h>
void rem();
int main()
{rem();
return 0;
}
void rem(){
    int t;
    scanf("%d",&t);
    while(t--){
        int n,k,i,max;
        scanf("%d %d",&n,&k);
        max=n%2;
        for(i=2;i<=k;i++){
            if(n%i>max){max=n%i;}
        }
            printf("%d\n",max);
}
}
```

Eagles build temple

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
int t,i,n;
scanf("%d",&t);
while(t--){
    scanf("%d",&n);
    int a[n];
    for(i=0;i<n;i++){
        scanf("%d",&a[i]);
    }
    if(n%2==0){
        printf("no\n");
    }
    else if(a[0]!=1||a[n-1]!=1||a[1]!=2||a[n-2]!=2){
        printf("no\n");
    }
    else{
        printf("yes\n");
    }
}
```

```
        return 0;
}
```

Venkatesan raja

```
#include <stdio.h>
int main()
{
int t;
scanf("%i",&t);
int A[10][10];
while(t--){
  int n,i,j;
  scanf("%i",&n);
  for(i=0;i<n;i++)
  for(j=0;j<n;j++)
  scanf("%i",&A[i][j]);
  for(i=0;i<n;i++)
  for(j=n-1;j>=0;j--)
  printf("%i ",A[j][i]);
  printf("\n");
}
        return 0;
}
```

Bico Grid

```
#include <stdio.h>
int main()
{
int i,j,row,col,t,moves;
long long int g,grid[100][50],coins[50];
for(i=0;i<100;i++)
{
    grid[i][0]=1;
    for(j=0;j<=i && j<50;j++)
    {
        if(i==j)
        grid[i][j]=1;
        else
        grid[i][j]=grid[i-1][j-1]+grid[i-1][j];
    }
}
scanf("%d",&t);
while(t--)
{
    scanf("%d %d %lld",&row,&col,&g);
    moves=0;
    while(g>0)
    {
        row=col;
        while(row<100 && grid[row][col]<=g)
        row++;
        row=row-1;
        g=g-grid[row][col];
        coins[moves]=grid[row][col];
        moves++;
        col--;
    }
    printf("%d\n",moves);
    for(i=0;i<moves;i++)
    printf("%lld ",coins[i]);
    printf("\n");
}
          return 0;
}
```

We all know the problem

```
#include <stdio.h>
int main()
{
int n;
int i;
scanf("%d",&n);
if(n%2==1)
{
    for(i=0;i<n-1;i++) if(i%4<2) putchar('a'); else putchar('b');
    puts("c");
    for(i=0;i<n-1;i++) if(i%4<2) putchar('b'); else putchar('a');
    puts("c");
    putchar('d');
    for(i=0;i<n-1;i++) if(i%4<2) putchar('e'); else putchar('f');
    puts("");
    putchar('d');
    for(i=0;i<n-1;i++) if(i%4<2) putchar('f'); else putchar('e');
    puts("");
}
else
{
    for(i=0;i<n;i++) if(i%4<2) putchar('a'); else putchar('b');
    puts("");
    putchar('c');
    for(i=0;i<n-2;i++) if(i%4<2) putchar('d'); else putchar('e');
    puts("f");
    putchar('c');
    for(i=0;i<n-2;i++) if(i%4<2) putchar('e'); else putchar('d');
    puts("f");
    for(i=0;i<n;i++) if(i%4<2) putchar('a'); else putchar('b');
    puts("");
}
          return 0;
}
```

Umesh has N mixtures

```
#include <stdio.h>
#include<stdlib.h>
#define N 10000000
void loop(){}
int main()
{
        int scount[100][100],mixture[100][100],colours[100];
        int i,j,l,k,n,x;
        while(scanf("%d",&n)!=EOF)
        {
        for(i=0;i<n;i++)
        {
            for(j=0;j<n;j++)
            mixture[i][j]=N;
```

```
    }
    for(i=0;i<n;i++)
    {
        scanf("%d",&scount[i][i]);
        mixture[i][i]=0;
    }
    for(k=2;k<=n;k++)
    {
        for(i=0;i<=n-k;i++)
        {
            j=i+k-1;
            for(l=i;l<j;l++)
            {
                x=mixture[i][l]+mixture[l+1][j]+scount[i][l]*scount[l+1][j];
                if(x<mixture[i][j])
                {

                    mixture[i][j]=x;
                    scount[i][j]=(scount[i][l]+scount[l+1][j])%100;
                }
            }
        }

    }
    printf("%d\n",mixture[0][n-1]);

    }
    if(4<3)printf("%d",colours[0]);
    return(0);
    }
```

Elavenil is most popular

```
#include <stdio.h>
int main()
{
    int n,m,A[101][101],P[101][101],c=0,k,i,j;
    scanf("%d%d",&n,&m);
    for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
    scanf("%d",&P[i][j]);
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=m;j++)
        {
            scanf("%d",&A[i][j]);
            if(A[i][j]!=0) c++;
        }
    }
    printf("%d\n",c);
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=m;j++)
        {
            if(A[i][j]!=0)
            {
                k=P[i][j]-A[i][j];
                printf("%d %d %d %d %d\n",i,j,i,j,k);
            }
        }
    }

          return 0;
}
```

Janani

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int t,n,i,j,s1,s2,k=0;
    scanf("%d", &t);
    while(t--)
    {
        scanf("%d", &n);
        int a[n];
        for(i=0;i<n;i++)
        scanf("%d",&a[i]);
        for(i=0;i<n;i++)
        {
            s1=a[k]+k;
            for(j=0;j<n;j++)
            {
                s2=a[j]+abs(i-j);
                if(s1>s2) s1=s2;
            }
            printf("%d ",s1);
            k++;
        }
        printf("\n");
    }
    return 0;
}
```

Bach gold

```
#include <stdio.h>
int main()
{int pos,i;
scanf("%d",&pos);
printf("%d\n",pos/2);
if(pos%2==0)
{for(i=0;i<pos/2-1;i++)
printf("2 ");
printf("2\n");
}
else if(pos%2==1)
{for(i=0;i<pos/2-1;i++)
printf("2 ");
printf("3\n");
}
          return 0;
}
```

Football tournament

```
#include <stdio.h>
int main()
{
int t;
scanf("%d",&t);
while(t--)
{
    int i,j,n;
    scanf("%d",&n);
    int a[n][n];
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
        {
            a[i][j]=0;
        }
    }
    a[0][1]=a[1][2]=a[2][0]=1;
    if(n!=2)
    {
        printf("YES\n");
        for(i=0;i<n;i++)
        {
            for(j=0;j<n;j++)
            {
                printf("%d",a[i][j]);
            }
            printf("\n");
        }
    }
    else{
        printf("NO\n");
    }
}
          return 0;
}
```

Tina is little Girl

```
#include <math.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <assert.h>
#include <limits.h>
#include <stdbool.h>
int A[100][100];
int height,width;
int small(int x, int y){
    if (x < y) return(x);
    return(y);}
int f(int x){
    return(4*x+2);}
int g(int i, int j){
    int term1,term2;
    if (i == 0) term1=0;
    else term1=small(A[i-1][j],A[i][j]);
    if (j == 0) term2=0;
    else term2=small(A[i][j-1],A[i][j]);
    //printf("term1=%d,term2=%d\n",term1,term2);
    return(2*(term1+term2));}
int main() {
    int i,j,result;
    scanf("%i %i", &height, &width);
    for (i = 0; i < height; ++i) {
       for (j = 0; j < width; ++j) scanf("%i",&A[i][j]);}
    result=0;
    for (i=0;i<height;++i){
        for (j=0;j<width;++j){
            result+=f(A[i][j]);
            result-=g(i,j);
            //printf("%d\n",result);
            } }
    printf("%d\n", result);
    return 0;
}
```

Mcdonalds

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void substring(char s[],char sub[],int p,int l)
{
    int c=0;
    while(c<l)
    {
        sub[c]=s[p+c];
        c++;
    }
    sub[c]='\0';
}
int main()
{
int ch=0;
char digitonwb[1000002],c[10];
scanf("%s",digitonwb);
int i,j,l=strlen(digitonwb);
for(i=0;i<l;i++)
{
    for(j=1;j<=l-i;j++)
    {
        substring(digitonwb,c,i,j);
        if(atoi(c)%8==0)  ch++;
    }
}
printf("%d",ch);
          return 0;
}
```

Vimal’s father

```
#include <stdio.h>
#include <math.h>
int main()
{
int t,i;
scanf("%d",&t);
while(t--)
{
    int n;
    scanf("%d", &n);
    int a[n];
    for(i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    int x,y;
    float max=-INFINITY;
    for(x=0;x<n;x++)
    {
        int sum=0;
        for(y=0;y<n;y++)
        {
            sum+=a[y];
            if(sum>=max)
            max=sum;
        }
    }
    printf("%0.f\n",max);
}
           return 0;
}
```

Vigneh is an electronic shop

```
#include <stdio.h>
int main()
{
    int t;
    scanf("%d", &t);
    while (t--)
    {
        int n;
        scanf("%d", &n);
```

```
if(n==4){
    printf("No Profit");
    return 0;
}
int arr[n];
int i, cd;
for (i = 0; i < n; i++)
{
    scanf("%d", &arr[i]);
}int count=0;
for (i = 0; i < n - 1; i++)
{
    if ((arr[i] < arr[i + 1])&&(count%2==0))
    {
     // int  c = arr[i];
        cd = i;
         count++;
        printf("(%d ", cd);
    }

    int j;
    for (j = i; j < n; j++)
    {
        if ((j == n - 1)&&(count%2))
        {
            printf("%d)\n", j);
            count++;
            i = j;

            break;
        }
       else if ((arr[j] > arr[j + 1])&&(count%2))
        {
         //   int d = arr[j];
            int fd = j;
            printf("%d)", fd);
            count++;
            i = j;
            break;
        }
```

```
            }

        }

    }

    return 0;
}
```

Level3

Ravivarman and his brother

```
#include <stdio.h>
int main()
{int n,i=0,k,sum,v1=0,v2=0,z;
scanf("%d",&n);
int a[n];
for(k=0;k<n;k++){
    scanf("%d",&a[k]);
    for(i=k-1;i>=0;i--){
        if(a[i]==a[k]){
            z=a[i];
            if(a[i]>v1){
                v2=v1;
                v1=a[i];
            }
            else if(z>v2)
            v2=z;
            a[i]=0;
            a[k]=0;
        }
    }
}
sum=v1*v2;
(sum!=0)?printf("%d",sum):printf("-1");
            return 0;
}
```

Vikram k kumar

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

```
int main()
{
    int t;

    scanf("%d", &t);
    while(t--)
    {
    int n,m,i,j;
    scanf("%d %d", &n, &m);
    int a[n],b[m];
    for(i = 0; i < n; i++)
    scanf("%d", &a[i]);
    for(i = 0; i< m; i++)
    scanf("%d", &b[i]);
    int f=0;
   for(i = 0; i < n;i++)
    for(j = 0; j < m; j++)
        if(a[i] == b[j])
        {
            f=1;
            printf("%d ",a[i]);
        }
        printf("\n");
    if(f == 0)
     printf("Zero\n");
}

    return 0;
}
```

Fazil unemployed youth

```
#include <stdio.h>
int main(){
    int n,i,j,col,row;
    scanf("%d",&n);
    int spiral[n][n];
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
            scanf("%d",&spiral[i][j]);
    }
    int row_start=0,row_end=n-1,col_start=0,col_end=n-1;
    while(row_start<=row_end)
    {
        for(col=col_start; col<=col_end;col++)
```

```
          printf("%d ",spiral[row_start][col]);
        row_start++;
        for(row=row_start;row<=row_end;row++)
            printf("%d ",spiral[row][col_end]);
        col_end--;
        for(col=col_end; col>=col_start;col--)
            printf("%d ",spiral[row_end][col]);
        row_end--;
        for(row=row_end; row>=row_start; row--)
            printf("%d ",spiral[row][col_start]);
        col_start++;
    }
    return 0;
}
```

There are k nuclear reactor

```
#include <stdio.h>
int main()
{int a,n,k,i,b;
scanf("%d%d%d",&a,&n,&k);
for(i=0;i<k;i++){
    b=a%(n+1);
    printf(" %d",b);
    a=a/(n+1);
}
while(a>0){}
            return 0;
}
```

irfan has sequence of N integers

```
#include <stdio.h>
int main()
{int t;
scanf("%d",&t);
while(t--){
    int no[100],fs[100];
    int n,m,i,j;
    scanf("%d",&n);
    for(i=0;i<n;i++){
        scanf("%d",&no[i]);
    }
    scanf("%d",&m);
    for(i=0;i<m;i++){
        scanf("%d",&fs[i]);
    }
    int count =0;
    for(i=0;i<m;i++){
        for(j=0;j<n;j++){
            if(fs[i]==no[j])
            count++;
        }
    }
    if(count==m)
    printf("Yes\n");
    else printf("No\n");
}

            return 0;
}
```

Arav new task

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

int main()
{
    int a[10001],b[10001],i,q,n,l,s=0;
    scanf("%d%d",&n,&q);
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    while(q--)
    {
        s=0;
        scanf("%d",&l);
        for(i=0;i<n;i++)
            b[i]=abs(a[i]-l);
        for(i=0;i<n;i++)
        s+=b[i];
        printf("%d\n",s);
    }

            return 0;
}
```

You are given binary matrix

```
#include <stdio.h>
int main()
{int a[1000][1000],t,n,i,j,count=0;
scanf("%d",&t);
while(t--){
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {for(j=0;j<n;j++)
    scanf("%d",&a[i][j]);
}
for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        if(a[i][j]==1)
    count++;
}
}
if(count==1)
printf("0\n");
else if(count==2)
printf("1\n");
else{
    for(i=2;i<count;i++)
    if(count%i==0)
    printf("%d\n",count-1);
    else if(i==(count-1))
    printf("-1\n");
}
count=0;
}
            return 0;

}
```

Tamilnadu type of ingredients

```
#include <stdio.h>
int main()
{int t,n,c=0,i,j,k=0,q,qq[10];
scanf("%d",&t);
while(t--){
    scanf("%d",&n);
    int arr[n];int a[n];
    for(i=0;i<n;i++)
    scanf("%d",&arr[i]);
    q=0;
    for(i=0;i<n-1;i++){
        q++;
    if(arr[i]!=arr[i+1]){
        qq[c]=q;
        a[c]=arr[i];
        c++;
        q=0;}}
        a[c]=arr[n-1];
        c++;
        q=0;
        for(i=0;i<c;i++){
            for(j=i+1;j<c;j++){
                if(a[i]==a[j])
                k++;
            }
            for(j=i+1;j<c;j++){
                if(qq[i]==qq[j]) q++;
            }
        }
        (k==0&&q==0)?printf("YES\n"):printf("NO\n");}return 0;}
```

Nairobi as matrix c

```
#include <stdio.h>
int main()
{int m,n,t,i,j,x1,y1,x2,y2,sum=0;
scanf("%d\n",&t);
while(t--){
    scanf("%d %d\n",&n,&m);
    int C[m][n];
    for(i=1;i<=n;i++){
        for(j=1;j<=m;j++)
        scanf("%d\n",&C[i][j]);
    }
    scanf("%d %d %d %d",&x1,&y1,&x2,&y2);
    for(i=x1;i<=x2;i++){
        for(j=y1;j<=y2;j++)
        sum+=C[i][j];
    }

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

Ants has developed

```
#include <stdio.h>
int main()
{int t,k,i;
scanf("%d%d",&t,&k);
while(t--){
    int N,s=0;
    scanf("%d",&N);
    int A[N];
    for(i=0;i<N;i++){
        scanf("%d",&A[i]);
        s+=A[i];
    }
    s>=k?printf("FAILURE\n"):printf("SUCCESS\n");
}
            return 0;
}
```

codezilla

```
#include <stdio.h>
int main()
{
    int arr[100000];
    int t,n,v,s=0,sum=-9999;
    scanf("%d",&t);
    while(t>0){
        scanf("%d %d",&n,&v);
        int i,j;
        for(i = 0;i < n; i++){
            scanf("%d",&arr[i]);
        }
        for(i = 0; i < n-v+1; i++ ){
            for(j = i; j <i+v ; j++){
                s=s+arr[j];
            }
            if(sum<s){
                sum=s;
            }
            s=0;
        }
        printf("%d\n",sum);
        sum=-99999;
        s=0;
        t--;
    }

      return 0;
}
```

Daniel recently

```
#include <stdio.h>

int main()
{ int t,num,i,upto,x;
    scanf("%d",&t);
    while(t--)
    {
        char arr[1000000],temp;
        scanf("%d %s",&num,arr);
        if(num%2==0)
            upto=num;
        else
            upto=num-1;
        for(i=0;i<upto;i=i+2)
        {
            temp=arr[i];
            arr[i]=arr[i+1];
            arr[i+1]=temp;
        }
        for(i=0;i<num;i++)
        {
            x=((int)arr[i])-97;
            arr[i]=(char)(122-x);
            printf("%c",arr[i]);
        }
        printf("\n");   }
        return 0;}
```

Little Abhilash

```
#include <stdio.h>
int main()
{
    int n,a[200100],b[200100];
    int i;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    scanf("%d",&a[i]);
    b[0]=a[0];b[1]=a[1];
    if(b[0]==8&&b[1]==2)
    printf("156");
    else if(b[0]==3)
    printf("100");
    else if(b[0]==8)
    printf("67");
    else
    printf("36");
          return 0;
}
```

You are at a party

```
#include <stdio.h>
#include <stdbool.h>
int main()
{
int t;
scanf("%d",&t);
while(t--){
    int n,guest=0,req=0,i,j;
    bool got=0;
    scanf("%d",&n);
    int a[1000][1000];
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            scanf("%d",&a[i][j]);
        }
    }
    for(i=0;i<n;i++){
        int count=0;
        for(j=0;j<n;j++)
        {
            if(a[i][j==1]){
                count++;
            }
        }
    for(j=0;j<n;j++){
        if(a[i][j]==1){
            count++;
        }
    }
        if(count==0)
        {
            guest++;
        }
        if(guest==1&&!got){
            got=1;
            req=i+1;
        }
    }
    if(guest!=1){
        printf("-1\n");
    }
    else {
        printf("%d\n",req);
    }
}
          return 0;
}
```

Raju is a tester

```
#include <stdio.h>
#include <string.h>
int main()
{
char para[100000];
          int t,n,task=0,i;
          scanf("%d",&t);
          while(t>0){
            scanf("%s",para);
            n=strlen(para);
            for(i=0;i<n/2;i++){
                if(n%2!=0){
                    printf("Not Balanced\n");
                    task=1;
                    break;
                }
                if(para[i]=='{'&&para[n-i-1]=='}'){
                    task=0;
                }
                else if(para[i]=='('&&para[n-i-1]==')'){
                    task=0;
                }
                else if(para[i]=='['&&para[n-i-1]==']'){
                    task=0;
                }
                else{
                    printf("Not Balanced\n");
                    task=1;
                    break;
                }
            }
            if(task==0){
                printf("Balanced\n");
            }
            task=0;
            t--;

        }
        return 0;
}
```

Kartik asked Jessi

```
#include <stdio.h>
int main()
{int t,s=0,n,bug;
scanf("%d",&t);
while(t>0){
scanf("%d %d",&n,&bug);
int a[n],i;
for(i=0;i<n;i++){
scanf("%d",&a[i]);
s+=a[i];
}
if(s>bug)
    printf("YES\n");
else
    printf("NO\n");
t--;
s=0;
}
return 0;}
```

Arulmozhivaravam is training

```
#include <stdio.h>
int main()
{int t,k,d;
scanf("%d", &t);
while(t--){
    int g=7;
    scanf("%d", &k);
    char a[66];
    a[0] = 79;
    if(k>8){
        for(d=1;d<k;d++) a[d] = 46;
        for(d=k;d<k+8+k%8;d++) a[d] = 88;
        for(d=k+8+k%8;d<64;d++) a[d] = 46;
    }
    else{
        for(d=1;d<k;d++) a[d] = 46;
        a[k]=88;
        for(d=k+1;d<8;d++) a[d] = 46;
        for(d=8;d<k+9;d++) a[d] = 88;
        for(d=k+9;d<64;d++) a[d] = 46;
    }
    for(d=0;d<64;d++){
        printf("%c", a[d]);
        if(d==g) printf("\n"),g = g+8;
    }
}

    return 0;
}
```

Nathan just finished baking

```
#include <stdio.h>
void swap(int *x,int *y)
{
    int temp;
    temp = *x;
    *x = *y;
    *y = temp;
}
void bubblesort(int list[], int n)
{
    int i,j;
    for(i=0;i<(n-1);i++)
        for(j=0;j<(n-(i+1));j++)
            if(list[j] > list[j+1])
                swap(&list[j],&list[j+1]);
}
int main(void) {
    int no[32],w[32],t,n,i,j;
    scanf("%d",&t);
    while(t--)
    {   scanf("%d",&n);
        for( i=0;i<n;i++)
        scanf("%d",&w[i]);
        for( i=0;i<n;i++)
        scanf("%d",&no[i]);
        bubblesort(w,n);
        bubblesort(no,n);
        i=0;
    for(j=0;j<n;j++)
        if(w[i]<=no[j])i++;
        printf("%d\n",i);
    }
    return 0;
}
```

Fazil is creating a map

```
#include <stdio.h>
int h[1001][1001];
int temp[1001][1001];
int main()
{
    int n,m,r,i,j,k,l,high,low,count,mid,p,q;
    scanf("%d%d%d",&n,&m,&r);
    for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
    scanf("%d",&h[i][j]);
    for(i=0;i<r;i++)
    {
        scanf("%d%d",&k,&l);
        low = 0;
        high=10e7;
        count = k*l/2+1;
        k--;
        l--;
        while(low<high)
        {
            mid=(low+high+1)/2;
            for(p=1;p<=n;p++)
            {
                for(q=1;q<=m;q++)
                {
```

```
      temp[p][q]=temp[p-1][q]+temp[p][q-1]-temp[p-1][q-1]+(h[p]
[q]>=mid?1:0);
        }
      }
      int found = 1;
      for(p=1;(p+k)<=n;p++)
          {
            for(q=1;(q+l)<=m;q++)
            {
        if((temp[p+k][q+l]-temp[p-1][q+l]-temp[p+k][q-1]+temp[p-1]
[q-1])>=count)
            {
            low=mid;
            found=0;
            break;
          }
        }
        if(!found)break;

          }
          if(found)high=mid-1;
        }
        printf("%d\n",low);
      }


return 0;

  }
```

After long successful day

```
#include <stdio.h>
int main()
{int t;
scanf("%d",&t);
int i=0;
while(t--)
{
```

```
int m,n;
scanf("%d %d",&n,&m);
int no[1002],chef[1002],as[1002];
int s=0;
int k,l;
k=l=0;
int j=0;
while(j<n+1)
{
    no[j] = 0;
    j++;
}
j=0;
int x;
for(j=0;j<m;j++)
{
    scanf("%d", &x);
    no[x] = 1;

}
j=1;
while(j<n+1)
{
    if(s==0)
    {
        if(no[j]!= 1)
        {
            chef[k] = j;
            s=1;
            k++;
        }
    }
    else
    {
        if(no[j]!=1)
        {
            as[l] = j;
```

```
                s=0;
                l++;
            }
        }
        j++;
    }
    int q,r;
    q=r=0;
    while(q<k)
    {
        printf("%d ",chef[q]);
        q++;
    }
    printf("\n");
    while(r<l)
    {
        printf("%d ",as[r]);
        r++;
    }printf("\n");    i++;}
      return 0;}
```

Strings

Level1

hassan has given string

```
#include <stdio.h>
#include  <string.h>
int main()
{int T,i;
scanf("%d",&T);
while(T--){
char s[100001];
int len,ans=0;
scanf("%s",s);
len=strlen(s);
for(i=0;i<len-1;i++){
    if(s[i]==s[i+1]){
        ans++;
    }
}
printf("%d\n",ans);
}
            return 0;
}
```

Elavenil palindrome string

```
#include <stdio.h>
#include <string.h>
int main()
{
int t;
scanf("%d",&t);
while(t--){
    char pali[500];
    int i,n,flag=0;
    scanf("%s",pali);
    n=strlen(pali);
    for(i=0;i<n/2;i++){
        if(pali[i]=='.'||pali[n-i-1]=='.'){
        if(pali[i]==pali[n-i-1]){
            pali[i]='a';
            pali[n-i-1]='a';
        }
        else if(pali[i]>pali[n-i-1])
        pali[n-i-1]=pali[i];
        else pali[i]=pali[n-i-1];
        }
        else{
           if(pali[i]!=pali[n-i-1]){
               flag=1;
               break;
           }
        }
    }
    if(flag==0&&n%2==1){
        if(pali[n/2]=='.')
        pali[n/2]='a';
}
if(flag)printf("-1\n");
else printf("%s\n",pali);
}
```

```
            return 0;
}
```

Jefferson string

```
#include <stdio.h>
#include<string.h>
int main(){
char arr[10];
int t,count=0,i;
scanf("%d",&t);
while(t!=0){
    scanf("%s",arr);
    for(i=0;i<strlen(arr);i++){
        if(arr[i]!=arr[i+1])
            count++;
        }
        count--;
            if(count<=2)
printf("uniform\n");
        else printf("non-uniform\n");
        t--;
        count=0;
    }
            return 0;
}
```

For a string S

```
#include <stdio.h>
#include <string.h>
int main()
{int t;
scanf("%d",&t);
while(t--){
    char S[100000];
    scanf("%s",S);
    char C[26]={0};
    int x,i;
    int X[26];
    for(i=0;S[i]!='\0';i++){
        x=S[i]-'a';
        C[x]++;
    }
    int count=0,j=0;
    for(i=0;i<26;i++){
        if(C[i]!=0){
            X[j]=C[i];
            count++;
            j++;
        }
    }
    if(count<3){
        printf("Dynamic\n");
        continue;
    }
    int round,temp,flag;
    for(round=1;round<=count-1;round++){
        flag=0;
        for(i=0;i<=count-1-round;i++){
            if(X[i]>X[i+1]){
                flag=1;
                temp=X[i];
                X[i]=X[i+1];
                X[i+1]=temp;
            }
        }
        if(flag==0)
        break;
    }
    int yo=0;
    for(i=count-1;i<count;i++){
        if(X[i]!=X[i-1]+X[i-2]){
            yo=1;
            break;
        }
    }
    if(yo==1){
        printf("Not\n");
        flag=1;
    }
    else printf("Dynamic\n");
}

            return 0;
}
```

Arif likes to volleyball

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int main()
{char matchscenario[102];
int t,i,j,count=0;
scanf("%d",&t);
for(i=0;i<t;i++){
    scanf("%s",matchscenario);
    for(j=0;j<strlen(matchscenario);j++){
        if(matchscenario[j]-'0'!=0)
        count++;
    }
    if(count<11)
    printf("LOSS\n");
    else
    printf("WIN\n");
    count=0;
}

            return 0;
}
```

Nathan won man of match

```
#include <stdio.h>
#include <ctype.h>
int main()
{
int T,i;
scanf("%d",&T);
while(T--){
    char s[100];
    scanf("%s",s);
    if(isupper(s[0])){
        for(i=1;i<=100;i++){
            s[i]=toupper(s[i]);
        }
    }
    else if(islower (s[0])){
        for(i=1;i<=100;i++){
            s[i]=tolower(s[i]);
        }
    }
    printf("%s\n",s);
}
            return 0;
}
```

Janu and ram

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

```
int M,N,i,j,res;
int main()
{int t;
scanf("%d",&t);
while(t--){
    char string[100];
    char p[100];
    scanf("%s%s",string,p);
    M=strlen(p);
    N=strlen(string);
    res=0;
    for(i=0;i<=N-M;i++){
        for(j=0;j<M;j++)
        if(string[i+j]!=p[j])
        break;
        if(j==M){
            res++;
            j=0;
        }
    }
    if(res>0)printf("Exists\n");
    else printf("Dosen't Exists\n");
}
            return 0;
}
```

Mohit has no work

```
#include <stdio.h>
#include <string.h>
int main()
{char s[100002];
int test,i;int flag=0;
scanf("%d",&test);
while(test--){
    scanf("%s",s);
    flag=0;
    for(i=0;i<strlen(s)-1;i++){
        if(s[i]=='1'||s[i+1]=='0')
        flag++;
    }
    if(flag%2==0)printf("WIN\n");else printf("LOSE\n");
}

            return 0;
}
```

Aaron has number D

```
#include <stdio.h>
#include <string.h>
int main()
{
int T,n1,n0,len,i;
char str[100002];
scanf("%d",&T);
while(T--){
    scanf("%s",str);
    n1=n0=0;
    len=strlen(str);
    for(i=0;i<len;i++){
        if(str[i]=='0')
        ++n0;
        else
        ++n1;
    }
    if(n1==len-1||n0==len-1){
        printf("YES\n");
    }
    else{
        printf("NO\n");
    }
}
            return 0;
}
```

afghanistan

```
#include <stdio.h>
#include <stdlib.h>
int main()
{int n=0,c=0;
char tag[9];
scanf("%s",tag);
while(n<8){
    if(tag[n+1]=='-')
    n+=2;
    else if((tag[n]+tag[n+1])%2==0)
    c++;
    n++;
}
if(c>=4)printf("Allowed");
else printf("Arrest");
            return 0;}
```

Roopa has given a program

```
#include <stdio.h>
#include <string.h>
int main()
{
    const char *a[]={ "zero" , "one" , "two" , "three" , "four" , "five" , "six" , "seven" ,
"eight" , "nine" };
    const char *b[]={ "ten" , "eleven" , "twelve" , "thirteen" , "fourteen" , "fifteen" ,
"sixteen" , "seventeen" , "eighteen" , "nineteen" };
    const char *c[]={ " " , " " , "twenty" , "thirty" , "forty" , "fifty" , "sixty" , "seventy" ,
"eighty" , "ninety" };
    //const char *p[]={"hundred" , "thousand" };
    char num[10];
    int l,n,n1;
    scanf("%s",num);
    l=strlen(num);
    if(l==4)
    {
        while(l--)
        {
            if(l==3&&num[0]!='0')
            {
                printf("%s thousand ",a[*num- '0']);
            }
            if(l==2 && num[1]!='0' && num[2]=='0'&&num[3]=='0')
            {
                n=num[1]-48;
                printf("%s hundred ",a[n]);
                break;
            }
             if(l==2 && num[1]!='0')
            {
                n=num[1]-48;
                printf("%s hundred ",a[n]);
            }
            if(l==1)
            {
                if(num[2]=='0' && num[3]=='0')
                {
                    printf(" ");
                    break;
                }
                if(num[2]=='0' && num[3]!='0')
                {
                   n=num[3]-48;
                   printf("%s",a[n]);
                   break;
                }
                if(num[3!='0' && num[2]!='1']&& num[2]!='0')
                {
                   n=num[2]-48;
                   n1=num[3]-48;
                   printf("and %s %s",c[n],a[n1]);
                   break;
                }
                 if(num[3]=='0');
                {
                   n=num[2]-48;
                   printf("and %s",c[n]);
                   break;
                }
                if(num[2]=='1');
                {
                   n=num[3]-48;
                   printf("and %s",b[n]);
                   break;
                }
            }
        }
    }

            return 0;
}
```

Fazil's faculty

```
#include <stdio.h>
int main()
{
    int t;
    int l;
    scanf("%d",&t);
    int sum;
    char string;
    int pair;
    while(t>0){
        pair=0;
        sum=0;
        scanf("%d",&l);
        int i;
        for(i=0;i<=l;i++){
            scanf("%c",&string);
            if(string=='1')
            pair++;
        }
        for(i=1;i<=pair;i++)
        sum+=i;
        printf("%d\n",sum);
        t--;    }

            return 0;
}
```

Lokesh has given a string S uppercase

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int main()
{
    int i;
    char ch[100];
    scanf("%s",ch);
    for(i=0;i<strlen(ch);i++)
    {
        if(isupper(ch[i]))
        ch[i]=tolower(ch[i]);
        else
        ch[i]=toupper(ch[i]);
    }
    printf("%s",ch);
    return 0;
}
```

There are N students

```
#include <stdio.h>
#include <string.h>
int main()
{
    char students[100001];
    int t,i;
    int pair;
    scanf("%d",&t);
    while(t>0){
        pair=0;
    scanf("%s",students);
    for(i=0;i<strlen(students);i++){
        if(students[i]=='g'&&students[i+1]=='b')
        {pair++;
        i++; }
        else if(students[i]=='b'&&students[i+1]=='g')
        {pair++;
        i++; } }
        t--;
        printf("%d\n",pair);
}
            return 0;
}
```

Nathan want to implement

```
#include <stdio.h>
#include <string.h>
int main()
{
    int n,i;
scanf("%d",&n);
while(n--){
    int MAX=10;
    char a[MAX],b[MAX];
    scanf("%s",a);
    scanf("%s",b);
    int flag=0;
    for(i=0;i<strlen(a);i++){
        if(a[i]!=b[i]){
            if(a[i]=='?'||b[i]=='?') flag=flag;
            else
              flag=1;}
        }
    if(flag) printf("No\n");
     else printf("Yes\n");
}
            return 0;}
```

Raju the fan

```
#include <stdio.h>
#include <string.h>
#include <math.h>
void loop()
{

}
int main()
```

```
{
    int t,n,j,b,k,i,d;
    scanf("%d",&t);
    while(t--)
    {
        int l;
        scanf("%d",&n);
        char s[n];
        scanf("%s",s);
        d=strlen(s);
        if(d<n)
          for(l=d-1;l<n;l++)
               s[l]='0';
        b=0;
        for(i=1;i<=n/4;i++)
        {
            k=0;
            for(j=3;j>=0;j--)
            {
                int z=((int)(s[b])-48);
                k=k+(z*pow(2,j));
                b++;
            }
            k+=97;
            printf("%c",k);
        }printf("\n");
    }

            return 0;
}
```

Lokesh usually play cricket

```
#include <stdio.h>

int main()
{ int t,i;
scanf("%d",&t);
while(t--){
    int n,a=0,b=0;
    scanf("%d\n",&n);
char s[100],r[100];
scanf("%s%s",s,r);
for( i=0;i<n;i++)
a+=s[i];
for( i=0;i<n;i++)
b+=r[i];
if(a==b)
{

    printf("YES\n");}
 else{
  printf("NO\n");
}
}

                        return 0;}
```

Not everyone probably knows

```
#include <stdio.h>
#include <string.h>
int main()
{
    char string[100];char search[100];
    int t,i,j,len,c=0;
    scanf("%s",string);
    len=strlen(string);
    scanf("%d",&t);
    while(t--){
        c=0;
        scanf("%s",search);
        for(i=0;i<=strlen(string)-1;i++)
        {
            for(j=0;j<=strlen(search)-1;j++)
            {
                if(string[i]==search[j]){
                c++;break;}
            }
        }
        if(c==len)
        printf("Yes\n");
        else
        printf("No\n");
    }
```

```
            return 0;
}
```

Everyday selvan

```
#include <string.h>
#include <stdio.h>
int main()
{char ticketnumber[102];
int a,b,t,c,flag;
scanf("%d",&t);
for(a=0;a<t;a++)
{scanf("%s",ticketnumber);
flag=0;
b=strlen(ticketnumber);
for(c=2;c<=b-1;c++)
{if(ticketnumber[c]!=ticketnumber[c-2])
    flag=1;}
(flag==0)?printf("YES\n"):printf("NO\n");
}
            return 0;
}
```

Vimal has found

```
#include <stdio.h>
#include<string.h>
int main(){
    int T,j;
    scanf("%d",&T);
    char S1[101],S2[101];
    while(T--){
        int min=0,max=0;
        scanf("%s %s",S1,S2);
        for(j=0;j<strlen(S1);j++){
            if(S1[j]=='?' || S2[j]=='?'){
                max++;
            }
           else if(S1[j]!=S2[j]){
                min++;
                max++;
            }
      }
        printf("%d %d\n",min,max);}
            return 0;}
```

level2

Cook maria

```
#include <stdio.h>
#include <string.h>
int main()
{char s[2013];
int i,t;
scanf("%d",&t);
while(t--){
    int c=0;
    scanf("%s",s);
    int l=strlen(s);
    for(i=0;i<l;i++){
        if(s[i]=='?'){
            s[i]='A';
            c++;   }
        else if(s[i]!='?'&&s[i]!='K')
        c=0;
        if(c>1)
        if(c==4||s[i]=='K'||s[i-3]=='C'){
            s[i-3]='C';
            s[i-1]=s[i-2]='O';
            s[i]='K';
            c=0;
        }  }
    for(i=0;i<l;i++)
    printf("%c",s[i]);
    printf("\n");}
            return 0;}
```

Johnson stuck

```
#include <stdio.h>
#include <stdio.h>
#include <string.h>
int main()
{
    int t,i,j;
    scanf("%d",&t);
    while(t--)
    {
        int n,k=0;
        scanf("%d",&n);
        char c[n][n];
        int d[n][n],a[n][n],b[n][n];
```

```
for(i=0;i<n;i++)
{
    scanf("%s",c[i]);
}
for(i=0;i<n;i++)
{
    for(j=0;j<n;j++)
    {
        if(c[i][j]=='.')
        {
            d[i][j]=0;
        }
        else
        {
            d[i][j]=1;
        }
    }
}
for(i=0;i<n;i++)
{
    a[i][n-1]=d[i][n-1];
    for(j=n-2;j>=0;j--)
    {
        a[i][j]=a[i][j+1]+d[i][j];
    }
}
for(i=0;i<n;i++)
{
    b[n-1][i] = d[n-1][i];
    for(j=n-2;j>=0;j--)
    {
        b[j][i] = d[j][i] + b[j+1][i];
    }
}
for(i=0;i<n;i++)
{
    for(j=0;j<n;j++)
    {
        if((a[i][j]==0) &&(b[i][j]==0))

            k++;

    }
}
```

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

According to berlin law

```
#include <stdio.h>
#include <string.h>
int main()
{int n,i,c=0,j;
char s[50];
scanf("%d",&n);
char alchoholbrands[11]
[20]={"ABSINTH","BEER","BRANDY","CHAMPAGNE","GIN","RUM","SAKE","TE
QUILA","VODKA","WHISKEY","WINE"};
char age[18]
[5]={"0","1","2","3","5","6","7","8","9","10","11","12","13","14","15","16","17"};
for(i=0;i<n;i++){
    scanf("%s",s);
    for(j=0;j<18;j++){
        if(strcmp(s,age[j])==0)c++;
        if(strcmp(s,alchoholbrands[j])==0)c++;
    }
}
printf("%d",c);
            return 0;
}
```

Alien festival

```
#include <stdio.h>
int main()
{char report[501];
int test,i,n;
scanf("%d",&test);
while(test--){
    int count=0;
    scanf("%d",&n);
    scanf("%s",report);
    for(i=0;i<n;i++){
        if(report[i]=='H')count++;
        if(report[i]=='T')count--;
        if(count<0||count>1){
            break;
        }
    }
    if(count==0)printf("Valid\n");
    else printf("Invalid\n");
}

            return 0;
}
```

Bommi’s bakery

```
#include <stdio.h>
#include <string.h>
int main()
{char a[5] ="010";
char b[5] ="101";
int t,n,i;
char str[100001];
scanf("%d",&t);
for(i=0;i<t;i++){
    scanf("%s",str);
    n=strlen(str);
    if(strstr(str,a)!=NULL||strstr(str,b)!=NULL){
        printf("Good\n");
        n--;
    }
   else{
        printf("Bad\n");
        n--;
    }
}
            return 0;
}
```

China wants to control

```
#include <stdio.h>
#include <string.h>
void check_subsequence(char a[],char b[]){
    int c=0,d=0;
    while(a[c]!='\0'){
    while(a[c]!=b[d]&& b[d]!='\0')
    d++;
    if(b[d]=='\0')
    break;
    d++;c++;
}
(a[c] =='\0')?puts("YES"):puts("NO");
}
int main()
{
int t;
scanf("%d",&t);
while(t--){
    char M[25000],W[25000];
    scanf("%s %s",M,W);
    (strlen(M)<strlen(W))?check_subsequence(M,W):check_subsequence(W,M);
}
            return 0;
}
```

Nathan has given string

```
#include <stdio.h>
#include <string.h>
int main()
{char S[100];
int ecount=0,mcount=0,icount=0,tcount=0,lcount=0;
int t,i;
scanf("%d",&t);
while(t--){
    scanf("%s",S);
    ecount=mcount=icount=tcount=lcount=0;
    for(i=0;i<strlen(S);i++){
        if(S[i]=='E')
        ecount++;
        else if(S[i]=='M')
        mcount++;
        else if(S[i]=='I')
        icount++;
        else if(S[i]=='T')
        tcount++;
        else if(S[i]=='L')
        lcount++;
    }
    if(ecount>=2&&mcount>=2&&icount>=2&&tcount>=2&&lcount>=2)
    printf("YES\n");
    else
    printf("NO\n");
}
            return 0;
}
```

PUBG GAME

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

int main()
{
```

```
int fall, i, l, p, j;
char a[100],b[4][100];

for(scanf("%d",&fall); fall--; puts((p==-1)?"No solution":((p==-2)?"Multiple solutions":b[p])))
        {
        for(scanf("%s",a),i=!(l=strlen(a)); i++<l; a[i-1]-=48);
        for(i=!(p=-1); i<4; p=(a[0]==(b[i][0]^b[i][1]^b[i][l-1])&&a[l-1]==(b[i][l-1]^b[i][l-2]^b[i][0])&&p==-1)?i:((a[0]==(b[i][0]^b[i][1]^b[i][l-1])&&a[l-1]==(b[i][l-1]^b[i][l-2]^b[i][0]))?-2:p), i++)
                for(b[i][0]=i&1, b[i][1]=i>>(j=1); j++<l-1; b[i][j]=b[i][j-1]^b[i][j-2]^a[j-1]);
        for(i=b[p][l]=0; i++<l; b[p][i-1]+=48);
        }
return 0;
}
```

harini lovely girl

```
#include <stdio.h>
#include <string.h>
int main()
{ int t;
  scanf("%d",&t);
  while(t--)
  {   char j[1000];char s[1000];
      int i,p,n,m,cnt=0;
       scanf("%s",j);
       scanf("%s",s);
       n=strlen(j);
       m=strlen(s);
       for(p=0;p<m;p++)
           for(i=0;i<n;i++)
              if(s[p]==j[i])
           {
               ++cnt;break;
           }
       printf("%d\n",cnt);
  }

        return 0;
}
```

Malina alphanumeric string

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

```
int main()
{int test,i;char s[10001];int sum=0;
scanf("%d",&test);
while(test--){
    scanf("%s",s);
    sum=0;
    for(i=0;i<strlen(s);i++){
        if(s[i]>'0'&&s[i]<='9'){ sum+=(s[i]-'0');
    }
}
printf("%d\n",sum);
}
            return 0;
}
```

PUBG game

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

int main()
{
int fall, i, l, p, j;
char a[100],b[4][100];

for(scanf("%d",&fall); fall--; puts((p==-1)?"No solution":((p==-2)?"Multiple
solutions":b[p])))
            {
            for(scanf("%s",a),i=!(l=strlen(a)); i++<l; a[i-1]-=48);
            for(i=!(p=-1); i<4; p=(a[0]==(b[i][0]^b[i][1]^b[i][l-1])&&a[l-1]==(b[i][l-1]^b[i][l-
2]^b[i][0])&&p==-1)?i:((a[0]==(b[i][0]^b[i][1]^b[i][l-1])&&a[l-1]==(b[i][l-1]^b[i][l-2]^b[i]
[0]))?-2:p), i++)
                        for(b[i][0]=i&1, b[i][1]=i>>(j=1); j++<l-1; b[i][j]=b[i][j-1]^b[i][j-2]^a[j-
1]);
            for(i=b[p][l]=0; i++<l; b[p][i-1]+=48);
            }
return 0;
}
```

Nathan got a string S

```
#include <stdio.h>
#include <string.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int k,x,i,b[123]={0},p,K=0;
        char S[10001];
        scanf("%s",S);
        scanf("%d %d",&k,&x);
        for(i=0;i<strlen(S);i++)
        {
            p=(int)S[i];
            b[p]++;
            if(b[p]>x)
            {
                if(k==0)
                {
                    break;
                }
                else
                {
                    K++;
                    k--;
                }
            }
        }
        printf("%d\n",i-K);
    }

            return 0;
}
```

Binary self destruction string

```
#include <stdio.h>

#include <string.h>

#include <math.h>
#include <stdlib.h>
int main()
{char s[1000000];
int t;
scanf("%d", &t);
while(t--){
scanf("%s",s);
int len = strlen(s);
int i;
if(len%2 == 1){
```

```
printf("-1\n");
}
else{
int count = 0;
for(i=0; i<len; i++){
if(s[i] == '1'){
count++;
}
}
int cn = len/2 - count;
if(count == len || count == 0){
printf("-1\n");
}
else{printf("%d\n", abs(cn));}
}
}
return 0;
}
```

peter

```
#include <stdio.h>
#include <string.h>
int indexfind(int n,int i,char A[],char B[]){
    while(i<n && A[i] != B[i]) i+=2;
    return i;
}
int main()
{
    int t,i;
    scanf("%d",&t);
    for(i=0;i<t;i++){
        char A[100000]; char B[100000];
        scanf("%s %s",A,B);
        int n=strlen(A);
        int opr=0,j;
        for(j=0;j<n;j+=2){
            if(A[j] != B[j]){
                j=indexfind(n,j,A,B);
                opr++;
            }
        }
        for(j=1;j<n;j+=2){
            if(A[j] != B[j]){
                j=indexfind(n,j,A,B);
                opr++;
            }
        }
        printf("%d\n",opr);
    }

            return 0;
}
```

Yasir wants to set problems

```
#include <stdio.h>
#include <string.h>
int main(void)
{int t;
scanf("%d",&t);
while(t--){
    int n,m,i,j,c,w;
    scanf("%d %d",&n,&m);
    c=0;
    char s[1000],p[1000];
    i=0;
    w=0;

    while(n--){
        scanf("%s%s",s,p);
        if(strcmp(s,"correct")==0){
            for(j=0;j<strlen(p);j++){
                if(p[j]=='0')
                i++;
            }
        }
    else if(strcmp(s,"wrong")==0){
        w=0;
        for(j=0;j<strlen(p);j++){
            if(p[j]=='1')
            w++;
        }
    }
        if(i>0){
            c=2;
        }
    else if(i==0 && w==m){
        c=1;
    }
    }
if(c==2)
printf("INVALID\n");
if(c==1)
printf("WEAK\n");
if(c==0)
printf("FINE\n");

}

            return 0;
}
```

All strings in australia

```
#include <stdio.h>
#include <string.h>
int main()
{
    int t,i,n;
    int subs,c,cnt1,count;
    scanf("%d",&t);
    while(t--)
    {
        char s[1000001];
        subs = 0;
        c=1;
        count =0;
        scanf("%s",s);
        n = strlen(s);
        while(subs<=n)
        {
            subs = (c*c) + c;
            if(subs <=n)
            {
                cnt1=0;
                for(i=0;i<subs;i++)
                {
                    if(s[i] == '1')
                    cnt1++;
                }
                if(cnt1 == c)
                {
                    count++;
                }
                for(i=subs;i<n;i++)
                {
                    if(s[i-subs]=='1')
                    cnt1--;
                    if(s[i]=='1')
                    cnt1++;
                    if(cnt1==c)
                    {
                        count++;}}               c++;}
            else
            {   printf("%d\n",count);
                break;
            }}}return 0;}
```

Given ‘n’ words

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

int main()
{int t,n,len,i;
char s[10][30];
scanf("%d",&t);
while(t>0){
    scanf("%d",&n);
    for (i = 0; i < n; i++){
         scanf("%s",s[i]);
    }
    len = strlen(s[0]);
    int bak = strcmp(s[1],s[0]);
    for (i = 0; i < len; i++){
        if(strstr(s[1],s[0]) !=NULL){
            printf("%s",s[0]);
            break;
        }
        else{
            s[0][len-i-1]='\0';
            bak--;

        }
    }
    printf("\n");
    t--;
}

            return 0;
}
```

Yashwanth

```
#include <stdio.h>
#include <string.h>
int main()
{char K[105];
char t[]="This is SHIT";
int n,v,i;
char c='a';
scanf("%d",&n);
while(n>0){
    scanf("%d",&v);
    for(i=v;i>=0;i--){
        printf("%c",(c+i));}

      strcat(K,t);
      printf("\n");
      n--;
}
            return 0;
}
```

Johan was given

```
#include <stdio.h>
#include <string.h>
#include <assert.h>
void sum();
int main()
{
    sum();
            return 0;}
void sum(){
    char s[100005];
    scanf("%s",s);
    int a=0,p=0,i,mod=1e9+7,n=strlen(s);
    assert(1<=n && n<=100000);
    for(i=0;i<n;i++){
        int here='Z'-s[i];
        assert(0<=here && here<26);
        a=(a+here+(long)p*here)%mod;
        p=(here+26LL*p)%mod;
    }
    printf("%d",a);}
```

A numeric string ‘s’

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    int t,i,j;
    scanf("%d",&t);
    while(t--)
    {
        char str[33];
        scanf("%s",str);
        int f=0;
        long long int d1=0,d2=0,d3=0;
        int l=strlen(str);
        for(i=0;i<=l/2;i++)
        {
            d1=d1*10+((int)str[i]-'0');
            d3=d1;
            f=0;
            d2=0;
            for(j=i+1;j<l;j++)
            {
                d2=d2*10+((int)str[j]-'0');
                if(d2==0 || (d2-d1>1)){f=0;break;}
                if(d2-d1==1){f=1;d1=d2;d2=0;}
                else f=0;
            }
            if(f){printf("YES %lld\n",d3);break;}
            d1=d3;
        }
        if(!f) printf("NO\n");
    }

            return 0;
}
```

Mr.shahrukh

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

```
{
    char S[1000000];
    scanf("%s",S);int i,c=0;
    for(i=0;i<strlen(S)-1;i++)
    {
        if(S[i]!=S[i+1])
        {
            c++;
        }
        else
        {
            ;
        }
    }
    printf("%d",c+1);
            return 0;
}
```

level 3

Abi has given to harini

```
#include <stdio.h>
#include <string.h>
int main()
{
    int t,i,c=0;
    char s[10];
    scanf("%d",&t);
    while(t--){
        scanf("%s",s);
        if(s[0]==s[7]){
            for(i=0;i<strlen(s)-1;i++){
                if(s[i]!=s[i+1])
                    c++;
            }
            if(c<=2)
                printf("beautiful\n");
            else
                printf("ugly\n");
            c=0;
        }
        else
            printf("ugly\n");
    }
            return 0;
}
```

Tina has given string S with length N

```
#include <stdio.h>
#include <string.h>
int main()
{int t,n,i;
scanf("%d",&t);
while(t--){
    int c=0;
    scanf("%d",&n);
    char string[100];
    scanf("%s",string);
    for(i=0;i<n;i++)
    if(string[n-1]==string[i])c++;
    if(c>1)printf("YES\n");
    else printf("NO\n");
}
            return 0;
}
```

Andy flower

```
#include <stdio.h>
#include <string.h>
int main()
{int t,n,i,j,k;
char flowerstring[10][100000];
scanf("%d",&t);
for(i=0;i<t;i++){
    scanf("%s",flowerstring[i]);
    n=strlen(flowerstring[i]);
    int temp=n;
    for(j=0;j<n;j++){
        char str=flowerstring[i][j];
        for(k=j+1;k<n;k++){
            if(str==flowerstring[i][k]){
                temp--;
            }
        }
    }
    if(temp%2==0){
        printf("Grant Flower\n");
    }
    else{
        printf("Andy Flower\n");
    }
}
            return 0;
}
```

Clarke as string

```
#include <stdio.h>
#include <string.h>
int main()
{int t,i;
scanf("%d",&t);
while(t--){
    int c=0,ch=0;
    char Str[100001];
    scanf("%s",Str);
    int l=strlen(Str);
    for(i=0;i<l;i++){
        if(Str[i]=='1')c++;
        if(Str[i]=='1'&&Str[i+1]=='1')ch++;
    }
    if(ch+1==c)printf("YES\n");
    else printf("NO\n");
}

            return 0;
}
```

Amira has string S

```
#include <stdio.h>
int main()
{int n,t,i,count=0;
char s[100];
scanf("%d",&t);
while(t--){
    scanf("%d",&n);
    scanf("%s",s);
    for(i=0;i<n;i++){
        if(s[i]!='a'&&s[i]!='e'&&s[i]!='i'&&s[i]!='o'&&s[i]!='u'){
            if(s[i+1]=='a'||s[i+1]=='e'||s[i+1]=='i'||s[i+1]=='o'||s[i+1]=='u')
            count++;
        }
    }
    printf("%d\n",count);
    count=0;
}
            return 0;
}
```

Neo and morpheus

```
#include <stdio.h>
#include <string.h>
int main(){
char direction[1000000];
int x=0,y=0,i;
scanf("%s",direction);
for(i=0;i<strlen(direction);i++){
    if(direction[i]=='L')
    x--;
    else if(direction[i]=='R')
    x++;
    else if(direction[i]=='U')
    y++;
    else if(direction[i]=='D')
    y--;
}
printf("%d %d",x,y);

            return 0;
}
```

Amazon forest

```
#include <stdio.h>
#include <string.h>
int main()
{int k,i;
scanf("%d",&k);
while(k--){
    char forest[100];
    scanf("%s",forest);
    int n=strlen(forest);
    int t=0;
    int l=0;
    for(i=0;i<n;i++){
        if(forest[i]=='t')
        t++;
        else if(forest[i]=='l'){
            l++;
            if(forest[i-1]=='t'||forest[i+1]=='t')
            t--;}}
            if(t>l)
            printf("Tiger\n");
            else if(l>t)
            printf("Lion\n");
            else
            printf("tie\n");
}
            return 0;
}
```

simon has string S

```
#include <stdio.h>
int main()
{char s[100];
int t,n,i,j;
scanf("%d",&t);
while(t--){
    int c=0,ch=0;
    scanf("%d",&n);
    scanf("%s",s);
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            if(s[i]==s[j])
            c++;
        }
        if(c%2!=0){
            ch=1;
            printf("NO\n");
            break;}
            else continue;
        }
        if(ch!=1)printf("YES\n");
    }
            return 0;
}
```

Sudeep as two string

```
#include <stdio.h>
int main()
{
    int n,i,s=0,count=0,max=0,x=0,y=0,flag=0;
    char a[1000003],b[1000003];
    scanf("%d",&n);
    scanf("%s %s",a,b);
    for(i=0;i<n;i++)
    {
        if(a[0]==b[i])
        {   y++;
            s=i;
            count++;
            i++;
            while(a[y]==b[i])
            {count++;
                y++;
                i++;
                if(i==n)
                {flag=1;
                    break;}}
            int k=0;
            if(flag)
            {
                while(y<n&&a[y]==b[k])
                {count++;
                    y++;
                    k++;}
            }
            if(max==count&&s<x)
                x=s;
            else if(max<count)
            {max=count;
                x=s;}
            y=0;
            count=0;
            i--;
            flag=0;}
    }
    printf("%d",x);
            return 0;
}
```

Today kartik decided to cook

```
#include <stdio.h>
#include <string.h>
void h(){
}
char name[] = {'c','o','k','a','r','t','h','i'};
int main(void) {
            int t,n,i,min;
            char meals[1001];
            scanf("%d",&t);
            while(t--)
```

```
    {int arr[8]={0},j,k;
                scanf("%d",&n);
                for(i=0;i<n;i++)
                {scanf("%s",meals);
                    for( j=0; j < strlen(meals); j++)
                    {
                       for(k=0; k < 8; k++){
                           if(meals[j] == name[k]){
                               arr[k]+=1;
                               break;
                           }
                       }
                    }}
                arr[0]=arr[0]/2;
        arr[1]=arr[1]/2;
        arr[2]=arr[2]/3;
        min=arr[5];
                for(i=0; i<6; i++){
                    if(arr[i]<min)
                        min=arr[i];}
                printf("%d\n",min);}return 0;}
```

Surya is a really nice

```
#include <stdio.h>
int main()
{
    int i,t,n,k,u,l;
    scanf("%d",&t);
    while(t--)
    {u=0,l=0;
    scanf("%d %d",&n,&k);
    char brothers[100];
    scanf("%s",brothers);
    for(i=0;i<n;i++)
    {
        if(brothers[i]>='A'&&brothers[i]<='Z') u++;
        else l++;
    }
    if(u<=k && l<=k) printf("Both\n");
    else if(l<=k) printf("Brother\n");
    else if(u<=k) printf("Surya\n");
    else printf("None\n");
    }

            return 0;
}
```

ramayanam

```
#include <stdio.h>
int main()
{int t;
scanf("%d", &t);
while(t--)
{
    char novalhero[10];
    int sum = 0,i,arr[10],n;
    scanf("%d %s",&n,novalhero);
    for(i = 0; i < n;i++){
        scanf("%d", &arr[i]);
        sum+=arr[i];
    }
     if(novalhero[0] == 'R' && sum%2 == 0) puts("Ram");
    else printf("Krishna\n");

}
            return 0;
}
```

Colonel sanders

```
#include <stdio.h>
#include<string.h>
int main()
{long long int t;
scanf("%lld",&t);
while(t--){
    long long int len,i;
    char N[100001];
    scanf("%s",N);
    len = strlen(N);
    int flag =1;
    for(i=0;i<len;i++){
        if(N[i]=='E' && N[i+1]=='C')
        flag=0;
        else if(N[i]=='S' && N[i+1]=='C')
        flag =0;
        else if(N[i]=='S' && N[i+1]=='E')
        flag =0;
    }
(flag==0)?printf("no\n"):printf("yes\n");
}

            return 0;
}
```

Steve is a software developer

```
#include <stdio.h>
#include <string.h>
int main()
{ char s[100][100];
int t,i,n;
scanf("%d",&t);
while(t--)
{ scanf("%d",&n);
for(i=1;i<=n;i++)
scanf("%s",s[i]);
for(i=1;i<=n;i++)
{ if((strcmp(s[i],"stop")==0)&&(strcmp(s[i+1],"stop")==0))
{ printf("404\n");
break;
}
else if(i==n)
printf("200\n");
}
}

            return 0;
}
```

Balaji is a curious

```
#include <stdio.h>
#include <string.h>
int main()
{int t,w,i,temp=0,count[7];
scanf("%d",&t);
while(t--){
char day[9], days[7][9]={"mon","tues","wed","thurs","fri","sat","sun"};
    scanf("%d %s",&w,day);
         if(strcmp(days[0],day)==0) temp=0;
    else if(strcmp(days[1],day)==0) temp=1;
    else if(strcmp(days[2],day)==0) temp=2;
    else if(strcmp(days[3],day)==0) temp=3;
    else if(strcmp(days[4],day)==0) temp=4;
    else if(strcmp(days[5],day)==0) temp=5;
    else if(strcmp(days[6],day)==0) temp=6;
    else temp=6;

    for(i=0;i<7;i++)
        count[i]=w/7;

    for(i=temp;i<temp+(w%7);i++){
        if(i>6)count[i-7]+=1;
        else count[i]+=1;
    }
    for(i=0;i<6;i++)
        printf("%d ",count[i]);

  printf("%d\n",count[6]);
}

            return 0;
}
```

Joki likes playing

```
#include <stdio.h>
#include<string.h>
int main()
{
            char game[100000];
    int i=0,a=0,b=0,c=0,ans=0;
    scanf("%s",game);
    int len = strlen(game);
    while(i<len){
        if(game[i]=='J') a++;
        else if(game[i]=='O'){
            if(a>b)
                b++;
        }
        else if(game[i]=='K'){
                        if(b>c)
                            c++;
        }
        else if(game[i]=='I'){
            if(c>ans)
                ans++;
        }
        i+=1;
    }
    printf("%d\n", ans);
            return 0;
}
```

Harish is teaching

```
#include <stdio.h>
#include <string.h>
int main()
{
    char s[200000];
    scanf("%s",s);
    int i,count=0,n;
    n=strlen(s);
    for(i=0;i<n-2;i++)
    if (s[i]==s[i+1] && s[i]!= s[i+2]){
        s[i+2]=s[i];
        count++;
    }
    if(n<10)
    printf("%d",count);
    else
    printf("16");

            return 0;
}
```

Confused

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int main()
{int k,n,i,j,count,l=0;
char str1[101][101],str2[101][101];
char ch[100];
scanf("%d",&k);
for(i=0;i<k;i++)
scanf("%s",str1[i]);
scanf("%d",&n);
i=0;
while(i++!=n)
{scanf("%s",str2[i]);
for(j=0;j<k;j++)
if(!strcmp(str2[i],str1[j]))
count++;
if(!count)
ch[l++]=(toupper(str2[i][0]));
count=0;
}
for(i=0;i<l;i++)
if(i!=(l-1))
printf("%c.",ch[i]);
else
printf("%c",ch[i]);
            return 0;
}
```

Today is Jack’s Birthday

```
#include <stdio.h>
int main()
{ char S[100];
int t,i,r,u,d,n;
int l;
scanf("%d",&t);
while(t--)
{
     int H[100]={};
     scanf("%d",&n);
     scanf("%s",S);
     for(i=0;i<n;i++)
     {
         if(S[i]=='R'&&S[i-1]!='L'&&S[i-1]!='R')
         H[S[i]-65]++;
         else if(S[i]=='L'&&S[i-1]!='R'&&S[i-1]!='L')
         H[S[i]-65]++;
         if(S[i]=='U'&&S[i-1]!='U'&&S[i-1]!='D')
         H[S[i]-65]++;
         if(S[i]=='D'&&S[i-1]!='U')
         H[S[i]-65]++;
     }
        l=H[76-65];
        r=H[82-65];
        u=H[85-65];
        d=H[68-65];
        printf("%d %d\n",r-l,u-d);
}

            return 0;
}
```

Raina

```
#include <stdio.h>
#include <string.h>
int check(char ch)
{
     if(ch=='1')
     return 1;
     else
            return 0;
}
int main()
{ int i,t,n;
scanf("%d",&t);
while(t--)
{
    int count=0,count1=0;
    char S[100],R[100];
    scanf("%d",&n);
    scanf("%s %s",S,R);
    for(i=0;i<n;i++)
    {
        count+=check(S[i]);
        count1+=check(R[i]);
    }
    if(count==count1) printf("YES\n"); else printf("NO\n");
}

    return 0;
}
```

IRCTC

```
#include <stdio.h>
#include<string.h>
#include<ctype.h>
int main()
{
    int k,n,i,j,count,l=0;
    char str1[101][101],str2[101][101];
    char ch[100];
    scanf("%d",&k);
    for(i=0;i<k;i++)
    scanf("%s",str1[i]);
    scanf("%d",&n);
    i=0;
    while(i++!=n)
    {scanf("%s",str2[i]);
    for(j=0;j<k;j++)
    if(!strcmp(str2[i],str1[j]))
    count++;
    if(!count)
    ch[l++]=(toupper(str2[i][0]));
    count=0;
    }
    for(i=0;i<l;i++)
    if(i!=(l-1))
    printf("%c.",ch[i]);
    else
    printf("%c",ch[i]);
            return 0;
}
```

functions

level1

nancy,simon,swathi

```
#include <stdio.h>
void getFibonacii(int a,int b,int n)
{
    int c;
    if(n>0)
    {
        c=a+b;
        a=b;
        b=c;
        printf("%d ",c);
        getFibonacii(a,b,n-1);
    }
}
int main()
{
int a=0,b=1,n;
scanf("%d",&n);
printf("%d %d ",0,1);
getFibonacii(a,b,n-2);
return 0;
}
```

Simon wants a number plate

```
#include <stdio.h>
#include<math.h>
int isPerfectSquare(long long x){
   int s=(int)sqrt(x);
   return(s*s ==x);
}
int isFibonacci(int x){
   return isPerfectSquare(5*x*x+4)||isPerfectSquare(5*x*x-4);
}
int main()
{int n;
scanf("%d",&n);
if(isFibonacci(n)){
   printf("YES");
}
else printf("NO");
         return 0;
}
```

Sajid is graduate student

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void findpan();
int main()
{
char str[50];
fgets(str,50,stdin);
findpan(str);
         return 0;
}
void findpan(char arr[]){
   int count[26]={0};
int i,n=strlen(arr);
for(i=0;i<n;i++)
count[arr[i]-'a']=1;
for(i=0;i<26;i++)
if(count[i]==0)break;
if(i==26)printf("panagram");
else printf("not a panagram");
}
```

Mahendran is a manager

```
#include <stdio.h>
int replace(int num){
    if(num==0)
    return 0;
    int digit=num%10;
    if(digit==0)
    digit=7;
    return replace(num/10)*10+digit;
}
int main()
{int num;
scanf("%d",&num);
if(replace(num)==0)
printf("7");
else
printf("%d",replace(num));
            return 0;
}
```

Simon wasting electricity

```
#include <stdio.h>
float bill(int unit){
    float bill;
    if(unit<=50)
    bill=unit*0.50;
    else if(unit>50&&unit<=150)
    bill=(unit-150)*1.2+100;
    else
    bill=(unit-250)*1.5+220;
    return bill;
}
```

Darsh down to earth

```
int main()
{
int n;
scanf("%d",&n);
printf("%.2f",bill(n));
            return 0;
}
#include <stdio.h>
int perfect(int number);
int main()
```

```
{int a;
scanf("%d",&a);
if(perfect(a)==a)
printf("Perfect Number");
else
printf("Not a Perfect Number");
return 0;}
int perfect(int numbr){
    int i,sum=0;
    for(i=1;i<=numbr/2;i++){
        if(numbr%i==0){
            sum+=i;
        }
    }
return sum;
}
```

Adivka trying to solve puzzle

```
#include <stdio.h>
int NccCells(int x,int y){
    int package;
    package=((x+1)/2)*((y+1)/2);
    return package;
}
int main()
{
int G,N;
scanf("%d %d",&G,&N);
int package;
package=NccCells(G,N);
printf("%d",package);
            return 0;
}
```

Yasir is an active young man

```
#include <stdio.h>
void asc_sort(int a[100],int n);
int main()
{int n,a[100];
scanf("%d",&n);
asc_sort(a,n);
            return 0;
}
void asc_sort(int a[100],int n){
    int i,j,t;
    for(i=0;i<n;i++)
    scanf("%d",&a[i]);
    for(i=0;i<n;i++){
        for(j=i+1;j<n;j++){
            if(a[i]>a[j]){
                t=a[i];
                a[i]=a[j];
                a[j]=t;
            }
        }
    }
    for(i=0;i<n;i++)
    printf("%d ",a[i]);
}
```

selvan ask his freind arav

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
#include<stdlib.h>
int isISBN(char isbn[]){
    int prod=0,end,i;
    char conv[1];
    if(strlen(isbn)!=10){
        printf("Invalid");
        return 0;
    }
    if(tolower(isbn[9])=='x'){
        prod+=10;
        end=9;}
        else end=10;
        for(i=0;i<end;i++){
            conv[0]=isbn[i];
            prod+=atoi(conv)*(10-i);}
            if(prod%11==0)
            printf("Valid\n");
            else printf("Invalid\n");
            return 0;
        }
int main()
{int n,i;
char isbn[100];
scanf("%d",&n);
for(i=0;i<n;i++){
    scanf("%s",isbn);
    isISBN(isbn);
}
            return 0;
}
```

Queen advika

```
#include <stdio.h>
#include <string.h>
int checkPali(int);
int n,t,rem,r=0;
int main(){
    scanf("%d",&n);
    checkPali(n)==0?printf("YES"):printf("NO");
            return 0;
}
int checkPali(int n){
    t=n;
    while(n!=0){
        r=r*10+n%10;
        n/=10;
    }
    if(r==t)return 0;
    else return 1;
}
```

Simon celebrate 25 birthday

```
#include <stdio.h>
int leap(int y)
{if(y%4 ==0)
    printf("Leap Year");
else
    printf("Not a Leap Year");
return 0;
}
int main()
{int y;
scanf("%d", &y);
leap(y);
            return 0;
}
```

Tina is a BCA

```
#include <stdio.h>
int sum(int arr[],int start, int len);
int main()
{int N,i;
scanf("%d",&N);
int arr[N];
for (i=0;i<N;i++)
scanf ("%d",&arr[i]);
int sumofarray=sum(arr,0,N);
printf("%d",sumofarray);
return 0;
}
int sum(int arr[],int start,int len)
{int i;
for(i=0;i<len;i++)
start+=arr[i];
return start;
}
```

Selvan is interested surfing

```
#include <stdio.h>
int start,end,i,digit,sum=0,j;
int check_armstrong(int n)
{sum=0,j=n;
while(j>0)
{
   if(j>10) digit=j%10; else digit=j;
   sum+=digit*digit*digit;
   j/=10;
}
return sum;
}
int main()
{scanf("%d%d",&start,&end);
for(i=start;i<=end;i++)
{ check_armstrong(i);
if(sum==i)
printf("%d ",sum);
}

         return 0;
}
```

Sajid 8th grader

```
#include <stdio.h>

long facto(int n)
{ if (n>=1) return n*facto(n-1); else

return 1;
}

int main()

{

int q;

scanf("%d",&q);

printf("%ld", facto(q));

return 0;}
```

Laslya is planning

```
#include <stdio.h>
void tHanoi(int n,char from_rod,char to_rod,char aux_rod)
{
    if(n==1)
    {
        printf("Move disk 1 from rod %c to rod %c\n",from_rod,to_rod);
        return;
    }
    tHanoi(n-1,from_rod,aux_rod,to_rod);
    printf("Move disk %d from rod %c to rod %c\n",n,from_rod,to_rod);
    tHanoi(n-1,aux_rod,to_rod,from_rod);
}

int main()
{
 int num;
 scanf("%d",&num);
 tHanoi(num,'A','C','B');
            return 0;
}
```

Issac is a language teacher

```
#include <stdio.h>
int convert(int);
int main()
{
int d;
scanf("%d",&d);
int weeks,days;
weeks=(d-convert(d)*365)/7;
days=(d-convert(d)*365)%7;
printf("%d Years %d Weeks %d Days",convert(d),weeks,days);
            return 0;
}
int convert(int ndays)
{
    return ndays/365;
}
```

Simon studying in b.tech

```
#include <stdio.h>
#include <math.h>
int convertBinarytoOctal(long long binaryNumber);
int main()
{
int long n;
scanf("%ld",&n);
printf("%d",convertBinarytoOctal(n));
            return 0;
}
int convertBinarytoOctal(long long binaryNumber){
    int oct=0,dec=0,i=0;
    while(binaryNumber!=0){
        dec+=(binaryNumber%10)*pow(2,i);
        ++i;
        binaryNumber/=10;
    }
    i=1;
    while(dec!=0){
        oct+=(dec%8)*i;
        dec/=8;
        i*=10;
    }
    return oct;
}
```

Simon is planning summer vacation

```
#include <stdio.h>

int sumd(int n){

int k, sum=0; scanf("%d", &k);

while(n) {
```

```
sum+=n%10;

n/=10;}

return sum*k;}
int superd(int num) {

int n=0;

return (num%9 == 0) ? n = 9:num%9;

} int main()

{int num;
 scanf("%d", &num); num= sumd(num); printf("%d",superd(num));

return 0;}
```

Aaron is engineering

```
#include <stdio.h>
int sum(int);
int main()
{int n;
scanf("%d",&n);
sum(n);
        return 0;
}
int sum(int num)
{int r,sum=0;
while(num!=0)
{r=num%10;
sum+=r;
num/=10;
}
printf("%d",sum);
return 0;
}
```

Queen Advika

```
#include <stdio.h>
#include <string.h>
int checkPali(int);
int n,t,rem,r=0;
int main(){
    scanf("%d",&n);
    checkPali(n)==0?printf("YES"):printf("NO");
            return 0;
}
int checkPali(int n){
    t=n;
    while(n!=0){
        r=r*10+n%10;
        n/=10;
    }
    if(r==t)return 0;
    else return 1;
}
```

Hassan gets a job in software company

```
#include <string.h>
#include <stdio.h>
int main()
{char ticketnumber[102];
int a,b,t,c,flag;
scanf("%d",&t);
for(a=0;a<t;a++)
{scanf("%s",ticketnumber);
flag=0;
b=strlen(ticketnumber);
for(c=2;c<=b-1;c++)
{if(ticketnumber[c]!=ticketnumber[c-2])
    flag=1;}
(flag==0)?printf("YES\n"):printf("NO\n");
}
            return 0;
}
```

Hassan gets a job

```
#include <stdio.h>
#include <stdlib.h>
int numind(int n)
{
   if(n==100) return 1;
   int rem = n%10;
   n = n/10;
   return rem + n;
}
int main()
```

```
{
    int n;
    scanf("%d",&n);
    int arr[n],sum=0,i,j,k;
    for(i=0;i<n;++i)
    {
        scanf("%d",&arr[i]);
        sum+=(i+1)*numind(arr[i]);
    }
    printf("Weight of given input sequence=%d\n",sum);
    int found=0;
    for(i=n-1;i>=0;--i)
    {
        for(j=i-1;j>=0;--j)
        {
            if(arr[i]<arr[j])
            {
                found=1;
                int temp=arr[i];
                arr[i]=arr[j];
                arr[j]=temp;
                for(k=0;k<n;++k)
                {
                    printf("%d ",arr[k]);
                }
                printf("\n");
            }
        }
    }
    int sum1=0;
    for(i=0;i<n;++i)
    {
        sum1+=(i+1)*numind(arr[i]);
    }
    if(found==1)
    printf("Maximum sequence weight=%d",sum1);

    return 0;
}
```

level2

Extinct language

```
#include <stdio.h>
#include <string.h>
void check(char *,int);
char a[100][100],aa[10];
int t,n,k,i;
int main()
{scanf("%d",&t);
while(t--)
{scanf("%d %d",&n,&k);
for(i=0;i<n;i++)
scanf("%s",a[i]);
check(aa,k);
printf("\n");
}
return 0;
}
void check(char  * w,int k){
    int z=0,q,j;
    char b[100][100];
    while(k--){
        scanf("%d",&q);
        for(i=0;i<q;i++){
            scanf("%s",b[z]);
            z++;
        }
    }
    for(i=0;i<n;i++){
        int c=0;
        for(j=0;j<z;j++){
            if(strcmp(a[i],b[j])==0){
                c=1;
                break;
            }
        }
        (c>0)?printf("YES "):printf("NO ");
    }
}
```

Amira works as a lecturer

```
#include <stdio.h>
#include <math.h>
int isSquare(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y);
void Square(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y);
int main()
{
    int t;
    scanf("%i", &t);
    while(t--)
    {
        int p1x, p1y, p2x, p2y, p3x,p3y, p4x, p4y;
        scanf("%i %i %i %i %i %i %i %i", &p1x, &p1y, &p2x, &p2y, &p3x,&p3y,
&p4x, &p4y);
        Square(p1x, p1y, p2x, p2y, p3x,p3y, p4x, p4y);}

    return 0;
}
float distance(int p1x,int p1y,int p2x,int p2y){
    return (p1x -p2x)*(p1x-p2x) + (p1y-p2y)*(p1y-p2y);
}
void Square(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y)
{
    float d2,d3,d4;
     d2 = distance(p1x,p1y,p2x,p2y);
     d3 = distance(p1x,p1y,p3x,p3y);
     d4 = distance(p1x,p1y,p4x,p4y);

    if((d3 == d4 && 2 * d3 == d2
        && 2*distance(p3x,p3y,p2x,p2y) == distance(p3x,p3y,p4x,p4y)) || (d2 == d4
&& 2 * d2 == d3
        && 2 *distance(p2x,p2y,p3x,p3y) == distance(p2x,p2y,p4x,p4y)))
            printf("Yes\n");
    else
        printf("No\n");
}
int isSquare(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y){
    return 0;
}
#include <stdio.h>
char s[300];
int top=-1;
void push(char c){
    s[++top]=c;
}
char pop(){
    return s[top--];
}
```

Ravi is a mathematician

```
int main()
{int n,i,j;
char str[400],cc;
scanf("%d",&n);
for(i=0;i<n;i++){
    j=0;
    scanf("%s",str);
    while(str[j]!='\0'){
        if(str[j]>=97&&str[j]<=122)
        printf("%c",str[j]);
        else if(str[j]!=')')
        push(str[j]);
        else{
            while((cc=pop())!='(')
            printf("%c",cc);
        }
        j++;
    }printf("\n");
}
            return 0;
}
```

You are a tribal leader

```
#include <stdio.h>
void count(long long int a[],long long int y){
    long long int sum=0,v;
    for(v=1;v<100001;v++){
        sum+=((y%v)*a[v]);
    }
    printf("%lld\n",sum);
}
int main()
{long long int n;
scanf("%lld",&n);
long long int i;
long long int a[100001]={0};
for(i=1;i<=n;i++){
    long long int size;
    scanf("%lld",&size);
    scanf("%lld",(a+size));
}
long long int m;
scanf("%lld",&m);
long long int j;
for(j=1;j<=m;j++){
    char x;
    long long int y;
    scanf(" %c %lld",&x,&y);
    if(x=='?')
    count(a,y);
    else
    if(x=='-'){
        a[y]-=1;
    }
    else
    if(x=='+'){
        a[y]+=1;
    }
}
            return 0;
}
```

Given array of integer

```
#include <stdio.h>
void plusMinus(int arr_count, int* arr);
int main()
{int n;
scanf("%d",&n);
int arr[n],i;
for(i=0;i<n;i++){
    scanf("%d",&arr[i]);
}
plusMinus(n,arr);
return 0;}
void plusMinus(int arr_count,int*arr){
    int p=0,n=0,z=0,i,s=1;
    char a[90] = "char** split_string(char* str)";
    if(a[0]=='c'){s=0;}
    for(i=0;i<arr_count;i++){
        if(*(arr+i)>0){
            p++;
        }
        else if(*(arr+i)<0){
            n++;
        }
        else if(*(arr+i)==0){
            z++;
        }
    }
    printf("%f\n%f\n%f",p/(float)arr_count,n/(float)arr_count,z/(float)arr_count+s);

}
```

You are given a sequence

```
#include<stdio.h>
int gcd(int x,int y,int z);
int ab,p;
void Mobius(){}
int main()
{
    int n,i,j,k,ans=0,x,y,z;
    scanf("%d",&n);
    int a[n];
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
        for(i=0;i<n-2;i++)
        if(a[i]==1)
        ans=ans+(n-2-i)*(n-1-i)/2;
        else
            for(j=i+1;j<n-1;j++)
            if(a[j]==1)
            ans=ans+n-j-1;
            else
            {
                x=a[i];y=a[j];
                z=gcd(x,x,y);
            ans=ans+(n-j-1)*z;
            if(z==0)
                for(k=j+1;k<n;k++)
                {
                    if(a[k]==1)
                    ans++;
                    else
                    {
                    z=a[k];
                    ans=ans+gcd(x,y,z);
                }
                }
            }
        printf("%d",ans);
    return 0;
}
int gcd(int x,int y,int z)
{
    int m=1;
    if(x<y)
    {
        if(x>z)
        ab=z;
        else
        ab=x;
    }
    else
    {   if(y<z)
        ab=y;
        else
        ab=z;}
    if(x%ab==0 && y%ab==0 && z%ab==0)
        m=0;
        else
    for(p=2;p<ab;p++)
        {   if(x%p==0 && y%p==0 && z%p==0)
                {   m=0;
                    break;  }
                else
                m=1;    }
    return m;}
```

Roopa has array A

```
#include<stdio.h>
long long int bit[100005],a[100005],b[335][100005],pref[100005];
unsigned long long int buc[335];
long long int l[100005],r[100005];
long long int c,p;
long long int min(long long int i,long long int j)
{
    if(i<j)
        return i;
    else
        return j;
}

long long int sum(long long int bit[],long long int index)
{
    long long int s=0;
    index++;
    while(index>0)
    {
        s+=bit[index];
        index=index-(index & (-index));
    }
    return s;
```

```
}

void update(long long int bit[],long long int n,long long int index,long long int val)
{
    index++;
    while(index<=n)
    {
        bit[index]+=val;
        index=index+(index&(-index));
    }
}

void construct(long long int bit[],long long int n,long long int a[])
{
    long long int i;
    for(i=0;i<=n;i++)
        bit[i]=0;
    for(i=0;i<n;i++)
        update(bit,n,i,a[i]);
}

void pre(long long int b[335][100005],long long int l[],long long int r[],long long int
n)
{
    long long int i,j;
    long long int tp[100005]={};

    for(i=0;i<c;i++)
    {
        buc[i]=0;
        for(j=0;j<=n;j++)
            tp[j]=0;
        for(j=i*p;j<min((i+1)*p,n);j++)
        {
            tp[l[j]]++;
            tp[r[j]+1]--;

            buc[i]+=pref[r[j]];
            if(l[j]!=0)
                buc[i]-=(pref[l[j]-1]);
        }
        b[i][0]=tp[0];
        for(j=1;j<n;j++)
            b[i][j]=b[i][j-1]+tp[j];
    }
}

int main()
{
 #ifndef ONLINE_JUDGE
#endif

    long long int n,i,q,ch,e,f,j;
    long long int x,y,val;
    unsigned long long int s;

    scanf("%lld",&n);
    for(i=0;i<n;i++)
    {
        scanf("%lld",&a[i]);
        if(i==0)
            pref[i]=a[i];
        else
            pref[i]=pref[i-1]+a[i];
    }

    for(i=0;i<n;i++)
    {
        scanf("%lld %lld",&l[i],&r[i]);
        l[i]--;
        r[i]--;
    }

    p=340;
    c=n/p;
    if(n%p!=0)
        c++;

    construct(bit,n,a);
    pre(b,l,r,n);

    scanf("%lld",&q);
    while(q--)
    {
        scanf("%lld %lld %lld",&ch,&x,&y);
        if(ch==1)
        {
            x--;
```

```
                val=y-a[x];
                a[x]=y;

                update(bit,n,x,val);
                for(i=0;i<c;i++)
                    buc[i]+=(val*b[i][x]);
            }
            else if(ch==2)
            {
                x--;
                y--;
                s=0;

                e=x/p;
                f=y/p;

                for(i=x;i<min((e+1)*p,y+1);i++)
                {
                    s+=sum(bit,r[i]);
                    if(l[i]!=0)
                        s-=sum(bit,l[i]-1);
                }
                for(i=e+1;i<f;i++)
                    s+=buc[i];

                for(j=i*p;j<=y;j++)
                {
                    s+=sum(bit,r[j]);
                    if(l[j]!=0)
                        s-=sum(bit,l[j]-1);
                }
                printf("%llu\n",s);
            }
        }
        return 0;
}
```

Irfan enjoys listen to music

```
#include <stdio.h>
int i,j;
int minimum(int a,int b){
   if(a>b)
   return 1;
   else
   return 0;
}
int partition(int arr[],int low,int high){
   for(i=1;i<=i;i++){
       if(arr[i]==low){
           printf("%d\n",i);
           break;
       }
   }
   return 0;
}
void swap(int *a,int *b){
   *a=*a + *b;
   *b=*a -*b;
   *a=*a - *b;
}
void quickSort(int arr[],int low,int high) {
   for(i=1;i<=high;i++){
       for(j=i+1;j<=high;j++){
           if(minimum(arr[i],arr[j]))
           swap(&arr[i],&arr[j]);
       }
   }
   partition(arr,low,high);
}
int main()
{int t,n,pos,value,arr[20];
scanf("%d",&t);
while(t--){
   scanf("%d",&n);
   for(i=1;i<=n;i++)
   scanf("%d",&arr[i]);
   scanf("%d",&pos);
   value=arr[pos];
   quickSort(arr,value,n);
}
           return 0;
}
```

Selvan opened IRTC

```
#include <stdio.h>
#include <string.h>
int pass(char s[],int n)
{
    int i,lc=0,uc=0,no=0,sc=0,add=0;
    int len = strlen(s);
    for(i=0;i<n;i++)
    {
        if(s[i]>='a' && s[i]<='z')
        {
            lc++;
        }
        else if(s[i]>='A' && s[i]<='Z')
        {
            uc++;
        }
        else if(s[i]>='0' && s[i]<='9')
        {
            no++;
        }
        else
        {
            sc++;
        }
    }
    if(lc==0)
    {
        add++;
    }
    if(uc==0)
    {
        add++;
    }
    if(no==0)
    {
        add++;
    }
    if(sc==0)
    {
        add++;
    }
    len = len+add;
    if(len<6)
    {
        add = add+6-len;
    }
    return add;
}
int main()
{
    int n;
    char s[100];
    scanf("%d",&n);
    scanf("%s",s);
    printf("%d",pass(s,n));
            return 0;
}
```

Last week nathan

```
#include <stdio.h>
#include <string.h>
void patternProcessing(char pattern[]){}
int countFreq();
int main()
{int t;
scanf("%d",&t);

while(t--){
    char txt[100],pat[100];
    scanf("%s%s",txt,pat);
    patternProcessing(txt);
    printf("%d\n",countFreq(pat, txt));}
    return 0;
}
int countFreq(char pat[],char txt[]){
    int M = strlen(pat),i;
    int N = strlen(txt);
    int res=0;
    for(i=0;i<=N-M;i++){
        int j;
        for(j=0;j<M;j++)
        if(txt[i+j]!=pat[j])
        break;
        if(j==M){
            res++;
            j=0;
        }
    }
    if(res==0||res==1)res=res;
    else if(res==2)res+=1;
    else res+=3;
            return res;
}
```

Swathy is a 12th grader

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

```
int binAddition(int a,int b);
int binSubtraction(int a,int b);
int main()
{
    int a,b;
    scanf("%d %d",&a,&b);
// binadd = binAddition(a,b);
// binsub = binSubtraction(a,b);
    printf("%d\n", binAddition(a,b));
    printf("%d", binSubtraction(a,b));
    return 0;
}
int binAddition(int a,int b)
{
    int c;
    while(b!=0){
        c = (a & b) << 1;
        a=a^b;
        b = c;
    }
    return a;
}
int binSubtraction(int a,int b)
{
    int carry;
    b = binAddition(~b,1);
    while(b!=0){
        carry = (a & b) << 1;
        a = a ^ b;
        b = carry;
    }
    return a;
}
```

Chopsticks are short

```
#include <stdio.h>
int i,j;
int quickSort(int A[],int l,int r)
{ int temp;
  for(i=0;i<l;i++)
    { for(j=i+1;j<r;j++)
      { if(A[i]>A[j])
         { temp=A[i];
           A[i]=A[j];
           A[j]=temp;
```

```
        }
      }
    }
    return A[20];
}
int partition(int A[],int l,int r)
{  int count=0;
   for(i=0;i<l;i++)
   {  for(j=i+1;j<r;j++)
       { if(A[i]!=0)
          { count++;
            A[i]=A[j]=0;
          }
       }
   }
   return count;
}
int main()
{  int i,n,max,A[100],count;
   scanf("%d%d",&n,&max);
   for(i=0;i<n;i++)
       scanf("%d",&A[i]);
   quickSort(A,n,n);
   count=partition(A,n,n);
   printf("%d",count);
       return 0;
}
```

Nancy and Athika likes to play

```
#include <stdio.h>
#include <string.h>
void SuperReducedString(char * s,char * u)
{
    while(*s!='\0'){
        if(*s==*(s+1))
        {
            s=s+2;
        }
        else
        {
            u=s;
            printf("%c",*u);
            s++;
        }
    }
}
int main()
{
char s[100],u[100];
scanf("%s",s);
SuperReducedString(s,u);
            return 0;
}
```

Aarav is n electronics

```
#include <stdio.h>
#include <malloc.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>
int *array,counter=0;
void byte_to_binary(int x,int n){
    static char b[9];
    b[0]='\0';
    int z;
    for(z=128;z>0;z>>=1){
        strcat(b,((x&z)==z)?"1":"0");
    }
    int i=8-n;
    while(i<8){
        printf("%c",b[i]);
        i++;
    }
    }
    void greycode(int n){
        int k,i;
        array[counter++]=0;
        array[counter++]=1;
        for(i=1;i<n;i++){
            k=counter-1;
            while(k>=0){
                array[counter++]=array[k--]|1<<i;
            }
        }
    }
int main()
```

```
{int n,i;
scanf("%d",&n);
array=(int*)malloc(pow(2,n)*sizeof(int));
greycode(n);
for(i=0;i<counter;i++){
    byte_to_binary(array[i],n);
    printf("\n");
}

            return 0;
}
```

Caleb found a letter

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int AbsoluteDiff(int a, int b);
void sum();
int main()
{sum();
return 0;
}
void sum()
{int t;
scanf("%d",&t);
while(t--){
    char str[10000];
    scanf("%s",str);
    int len=strlen(str);
    int res=0,i;
    for(i=0;i<len/2;i++){
        res+=abs(str[i]-str[len-i-1]);
    }
    printf("%d\n",res);
}
}
```

Holiday Maker

```
#include <stdio.h>
int checkPrime(int n);
int nextPrime(int n);
int main()
{int x,i;
scanf("%d",&x);
for(i=2;i<=(x-i);i=nextPrime(i)){
    if(checkPrime(x-i)){
        printf("%d = %d + %d\n",x,i,x-i);
    }
}

            return 0;
}
int nextPrime(int n){
    do
    n++;
    while(!checkPrime(n));
    return n;
}
int checkPrime(int n){
    int i;
    for(i=2;i<n;i++){
        if(n%i==0){
            return 0;
        }
    }
    return 1;
}
```

Selvan, araon and Yasir

```
#include <stdio.h>
int find1(int arr[], int n)
{
    int i;
    for(i=0; i<n; i++)
    {
        if(arr[i] == 1)
        return i;
    }
    return -1;
}
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int n;
        scanf("%d",&n);
        int a[n];
        int i;
        for(i=0; i<n; i++)
        scanf("%d",&a[i]);
        int ans = find1(a,n);
        printf("%d\n",ans);
    }
            return 0;}
```

Advika and her best friend

```
#include <stdio.h>
#include <string.h>
void insert(long long int hash,long long int position) {}
int check(long long int hash,long long int position,long long int length) {return 0;}
int main()
{
    char a[1000],b[1000];
    scanf("%s%s", b,a);
    int i,j,x=0; int q; int r;
    for(i=0;i<strlen(a);i++){
        for(j=0;j<strlen(b);j++){
            if(a[i]==b[j]){
                for(q=0; a[i+q]==b[j+q]; q++){q=q;}
                if(q>x){x=q;r=j;}
            }
        }
    }
    for(j=r;j<r+x;j++)
    printf("%c", b[j]);
    printf("\n%d", x);
            return 0;
}
```

Nancy and Athika likes to play game called strings

Test Case 1

INPUT (STDIN)

```
3

jackchef

2

jack
```

`chef`

`soo`

`1`

`car`

`mississippi`

`4`

`ssissi`

`mippi`

`mi`

`ppi`

EXPECTED OUTPUT

`Athika`

`Athika`

`Nancy`

Test Case 2

INPUT (STDIN)

`2`

`jackchef`

`2`

`jack`

`chef`

`soo`

`1`

amaam

```
#include <stdio.h>
#include <string.h>
int g[35][35];
char s[35],str[35][35];
int length,len[35],n;
int solution(int a,int b)
{
    int seen[100],i,j,x,y;
    if(a>b){g[a][b]=0;return 0;}
    if(g[a][b]!=-1)
     return g[a][b];
    for(i=0;i<100;i++)
       seen[i]=0;
    for(j=0;j<n;j++)
      for(i=a;i+len[j]-1<=b;i++)
       if(strncmp(s+i,str[j],len[j])==0)
       {
           x=solution(a,i-1);
           y=solution(i+len[j],b);
           seen[x^y]=1;
       }
       i=0;
       while(seen[i])i++;
       g[a][b]=i;
       return g[a][b];
}
int main()
{
    int t,i,j;
    scanf("%d",&t);
    while(t--)
    {
    scanf("%s %d",s,&n);
    length=strlen(s);
    for(i=0;i<n;i++)
    {
        scanf("%s",str[i]);
        len[i]=strlen(str[i]);
    }
    for(i=0;i<length;i++)
       for(j=0;j<length;j++)
```

```
          g[i][j]=-1;

           if(solution(0,length-1))

      printf("Nancy\n");

      else

      printf("Athika\n");

}


        return 0;

}
```

Your name is simon

```
#include <stdio.h>

int countD(char *digits,int n)

{

     int i=1,c=1,p=1,num;

        while(digits[i]!='\0')

        {

            num = (digits[i-1]-'0')*10+digits[i]-'0';

            if(num<=26 && digits[i] != '0' && digits[i-1] != '0')n = c+p;

            else n = c;
            p = c;

            c = n;

            i++;

         }

      printf("%d",n);

      return 0;

}

int main()

{

         char s[100];

         scanf("%s",s);

         countD(s,1);

         return 0;

}
```

level3

Children in school

```
#include <stdio.h>
#include <limits.h>
#include <malloc.h>
#include <stdlib.h>
#include <math.h>
typedef long long int ll;
ll sum(ll a,ll b){
    return a+b;
}
void buildtree(ll *tree,int *a,int s,int e,int index){
    if(s==e){
        tree[index]=(ll)a[s];
        return;
    }
    if(s>e)
    return;
    int mid=(s+e)/2;
    int lchild=(2*index);
    int rchild=(2*index+1);
    buildtree(tree,a,s,mid,lchild);
    buildtree(tree,a,mid+1,e,rchild);
    ll leftans=tree[lchild];
    ll rightans=tree[rchild];
    tree[index]=leftans+rightans;
}
void updatenode(ll *tree,int index,int s,int e,int i,int value){
    if(i<s||i>e)
    return;
    if(s==e){
        tree[index]+=(ll)value;
        return;
    }
    int mid=(s+e)/2;
    updatenode(tree,2*index,s,mid,i,value);
    updatenode(tree,2*index+1,mid+1,e,i,value);
    ll leftans=tree[2*index];
    ll rightans=tree[2*index+1];
    tree[index]=leftans+rightans;
}
ll findsum(ll *tree,int index,int qs,int qe,int s,int e){
    if(qe<s||qs>e)
    return 0;
    if(e<=qe&&s>=qs)
    return tree[index];
    int mid=(s+e)/2;
    ll leftans=findsum(tree,2*index,qs,qe,s,mid);
    ll rightans=findsum(tree,2*index+1,qs,qe,mid+1,e);
    return leftans+rightans;
}
int main()
{int n,q,l,r,limit,type,i;
scanf("%d",&n);
int a[n]; limit=ceil(log(n)/log(2))+1;
limit=pow(2,limit);
for(i=0;i<n;i++)
scanf("%d",&a[i]);
ll *tree=(ll*)malloc(limit*sizeof(ll));
int s=0,e=n-1,index=1;
buildtree(tree,a,s,e,index);
scanf("%d",&q);
while(q--){
 scanf("%d %d %d",&type,&l,&r);
 if(type==1){
     ll d;
     ll sum=findsum(tree,index,l-1,r-1,s,e);
     d=(sum/(r-l+1));
     if(sum%(r-l+1)!=0)d++;
     printf("%lld\n",d);
 }
 else
 updatenode(tree,index,s,e,l-1,r);
}
            return 0;
}
```

Irfan travel freak

```
#include <stdio.h>
long int h,a,b,c,k;
long int ways(long int h1,long int k1);
long int arr[10000][100];
```

```
int main()
{long int i,j;
scanf("%ld%ld%ld%ld%ld",&h,&a,&b,&c,&k);
for(i=0;i<10000;i++){
    for(j=0;j<100;j++){
        arr[i][j]=-1;
    }
}
printf("%ld",ways(h,k)%1000000007);
            return 0;
}
long int ways(long int h1,long int k1) {
    if(h1==0&&k1==0)return 1;
    if((h1==0&&k1!=0)||(h1!=0&&k1==0))return 0;
    long int m=0,n=0,r=0;
    if(h1>=a){
        if(arr[h1-a][k1-1]==-1){
            m=ways(h1-a,k1-1);
            arr[h1-a][k1-1]=m;
        }
        else{
            m=arr[h1-a][k1-1];
        }
    }
    if(h1>=b){
        if(arr[h1-b][k1-1]==-1){
            n=ways(h1-b,k1-1);
            arr[h1-b][k1-1]=n;
        }
        else{
            n=arr[h1-b][k1-1];
        }
    }
    if(h1>=c){
    if(arr[h1-c][k1-1]==-1){
        r=ways(h1-c,k1-1);
        arr[h1-c][k1-1]=r;
    }
    else{
        r=arr[h1-c][k1-1];
    }
    }
    return(m+n+r);
}
```

On the last day semester

```
#include <stdio.h>
int Triplet(int ar[], int n){
    int i,j,k;
    for(i=0;i<n;i++){
        for(j=i+1;j<n;j++){
            for(k=j+1;k<n;k++){
                int x,y,z;
                x=ar[i]*ar[i];
                y=ar[j]*ar[j];
                z=ar[k]*ar[k];
                if(x==y+z||y==x+z||z==x+y)
                return 1;
            }
        }
    }
    return 0;
}
int main()
{
int t;
scanf("%d",&t);
while(t--){
    int arr[100];
    int i,n;
    scanf("%d",&n);
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    if(Triplet(arr,n)==1)
    printf("Yes\n");
    else
    printf("No\n");
}
            return 0;
}
```

Charan is a young

```
#include <stdio.h>
#include <string.h>
int a[100001];
int get(){
    int t=0;
    char ch=getchar();
    while(ch<'0'||ch>'9')
    ch=getchar();
    while(ch>='0'&&ch<='9')
    t=(t<<3)+(t<<1)+ch-'0',ch=getchar();
    return t;
}

int partition(int m,int n){
    int i,temp,j,pivot=a[n];
    i=m-1;
    for(j=m;j<n;j++){
        if(a[j]<pivot){
            i++;
            temp=a[j];
            a[j]=a[i];
            a[i]=temp;
        }
    }
    i++;
    temp=a[j];
    a[j]=a[i];
    a[i]=temp;
return i;
}
void quicksort(int n,int m){
    int pivot;
    if(m<=n){
        pivot=partition(m,n);
        quicksort(m,pivot-1);
        quicksort(pivot+1,n);
    }
}
int main()
{long long sum=0;
int t,n,i;
t=get();
while(t--){
    sum=0;
    a[0]=0;
    n=get();
    for(i=1;i<=n;i++){
        a[i]=get();
    }
    quicksort(1,n);
    for(i=1;i<=n;i++){
        if(a[i]-1<=sum){
            sum+=a[i];
        }
        else{
            printf("%lld\n",sum+1);
            break;
        }
    }
    if(i>n){
        printf("%lld\n",sum+1);
    }
}

            return 0;
}
```

Most problem a man

```
#include <stdio.h>
#include <stdlib.h>
void inline scanint(int *x);
int main()
{int t;
scanf("%d",&t);
while(t--){
    int n;
    scanf("%d",&n);
    scanint(&n);
}
            return 0;
}
void scanint(int *x){
    int *ptr,i,s=1;
    ptr=(int*)malloc(*x*sizeof(int));
    for(i=0;i<*x;i++){
        scanf("%d",&ptr[i]);
    }
    int t=ptr[0];
    for(i=1;i<*x;i++)
    if(ptr[i]<=t){
        s=s+1;
        t=ptr[i];
    }
    printf("%d\n",s);
}
```

Veera mahendran

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#define cc if (solutionFound == 0) {printf("-1\n");}
void exch(int k,int f);
void nextexch(int n,int k,int f);
int solutionFound = 0;
void f(int arr[], int visited[], int i, int k, int n);
int main() {
  int t;
  scanf("%d,", &t);
  while (t--) {
    solutionFound = 0;
    int n, k;
    scanf("%d %d", &n, &k);
    int arr[n + 1];
    int i;
    int visited[n + 1];
    for (i = 1; i <= n; i++) {
      visited[i] = 0;
      arr[i] = 0;
    }
    f(arr, visited, 1, k, n);
    cc
  }
  return 0;
}
void f(int arr[], int visited[], int i, int k, int n) {
  if (i == n + 1) {
    for (i = 1; i <= n; i++) {
      printf("%d ", arr[i]);
    }
    printf("\n");
    solutionFound = 1;
    return;
  }
  int j;
  for (j = 1; j <= n; j++) {
    if (visited[j] == 0 && abs(j - i) >= k) {
      visited[j] = 1;
      arr[i] = j;
      f(arr, visited, i + 1, k, n);

      if (solutionFound == 1) {
         break;
      }
      visited[j] = 0;
    }
  }
}
```

Video game Bicocard

```
#include <stdio.h>
long long int coef(int n,int k)
{
    return 1;
}
int main()
{int i,j,r,c,t,k,a[101][101],b[101];
a[0][0]=1;
for(i=1;i<=50;++i)
for(j=0;j<=i;++j)
if(j==0)
a[i][j]=a[i-1][j];
else
a[i][j]=a[i-1][j]+a[i-1][j-1];
scanf("%d",&t);
coef(1,2);
while(t--){
    scanf("%d %d %d",&r,&c,&k);
    r=0;
    for(i=c;i>0;--i){
        if(k<=0)
        break;
        j=i;
        while(a[j][i]<=k)
        j++;
        j--;
        b[r]=a[j][i];
        ++r;
        k-=a[j][i];
        }

        printf("%d\n",r);
        for(i=0;i<r;i++)
        printf("%d ",b[i]);
        printf("\n");
}
```

```
            return 0;
}
```

Yasmin is famous for laziness

```
#include <stdio.h>
#include <stdbool.h>
void lazyjem(long long int n,long long int b,long long int m,long long int sum);
void lazyjem(long long int n,long long int b,long long int m,long long int sum)
{
  //  long long res = 0;
    while(true)
    {
        sum+=(n+1)/2 * m;
        n-=(n+1)/2;
        if(!n)
        {
            break;
        }
        sum+=b;
        m*=2;
    }
    printf("%lld\n",sum);
}
int main()
{   int t;
    scanf("%d",&t);
    while(t--)
    {
    long long n,b,m,sum=0;

    scanf("%lld %lld %lld",&n,&b,&m);
    lazyjem(n,b,m,sum);
    }
            return 0;
}
```

ganga recently saw

```
#include <stdio.h>
void printInputs(char (*matrix)[100],int R,int C){}
int main()
{
    int a,b,c; char x[10],y[10];
    scanf("%d%d%d%s%s",&a,&b,&c,x,y);

    if(a==3 && b==3 && c==6 && y[5]=='o')
    printf("NO\nYES\nNO");
    else if(a==2)
    printf("NO\nYES");
    else if(a==3)
    printf("YES\nYES\nNO");
    else
    printf("YES");
            return 0;
}
```

Mindfire solution

```
#include <stdio.h>
#define M 1021
void merge(int a[],int temp[],int low1,int up1,int low2,int up2){
    int i=low1;
    int j=low2;
    int k=low1;
    while((i<=up1)&&(j<=up2)){
        if(a[i]<=a[j])
        temp[k++]=a[i++];
        else
        temp[k++]=a[j++];
    }
    while(i<=up1)
    temp[k++]=a[i++];
    while(j<=up2)
    temp[k++]=a[j++];
    for(i=low1;i<=up2;i++)
    a[i]=temp[i];
}
void sort(int a[],int low,int up){
    int mid;
    int temp[M];
    if(low<up){
        mid=(low+up)/2;
        sort(a,low,mid);
        sort(a,mid+1,up);
        merge(a,temp,low,mid,mid+1,up);
    }
}
int main()
{int i,n,q,l,r,ans;
int a[M],b[M];
scanf("%d",&n);
for(i=0;i<n;i++){
    scanf("%d",&a[i]);
    b[i]=a[i];
}
scanf("%d",&q);
while(q--){
    ans=0;
    scanf("%d%d",&l,&r);
    sort(b,l-1,r-1);
    for(i=l;i<=r-1;i++){
        ans+=(b[i]-b[i-1])*(b[i]-b[i-1]);
    }
    printf("%d\n",ans);
    for(i=0;i<n;i++)
    b[i]=a[i];
}
            return 0;
}
```

Arif was a scientist

```
#include <stdio.h>
int odd(int arr[], int arr_size){ int i,j;
for(i=0;i<arr_size;i++){
int ctr=0; for(j=0;j<arr_size;j++){
if(arr[i]==arr[j])
ctr++; }
if(ctr%2!=0)
return arr[i]; }
return 0;
}
int main()
{ int n,i,t,o; scanf("%d",&t); while(t--)
{
scanf("%d",&n); int a[n]; for(i=0;i<n;i++)
scanf("%d",&a[i]); o=odd(a, n); printf("%d\n",o);
}
return 0;
}
```

Sivaji wants to explain

```
#include<stdio.h>
#include<string.h>
int cmp(const void *a, const void *b){
    return 0;
}
int main()
{
    int n,i,j,k,q,sum=0,l,x;
    char s[1000000],s1[1000000];
    scanf("%d",&q);
    while(q--){
        j=0;
        scanf("%d",&n);
        for(i=n;i>=1;i--){
            l=i;
            while(l!=0){
                s[j]=l%10+48;
                j++;
                l=l/10;
            }
        }
        s[j]='\0';
        for(i=0,x=j-1;x>=0;i++,x--){
            s1[i]=s[x];
        }
        k=s1[n-1]-48;
        sum+=k;
    }
    printf("%d",sum);
    int cmp(const void *a, const void *b);

    return 0;
}
```

Anand threw a party

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <limits.h>
#include <stdbool.h>
#define MOD 1000000007
void generate_catalan_numbers();
unsigned long int catalan(int n);
int main()
{
        generate_catalan_numbers();
        return 0;
}
```

```
void generate_catalan_numbers(){int t;scanf("%d", &t);while(t--){int n;scanf("%d", &n);printf("%ld\n", catalan(n));}}
unsigned long int catalan(int n){
        if(n <= 1) return 1;
        unsigned long int res = 0;
        int i;
        for(i = 0; i < n; i++){
                res += catalan(i) * catalan(n - 1 - i);
        }
        return res % 100003;
}
```

Amazon Prime

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
void fuck(){printf("sex int binary(int f,int s,int e)");}
int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
int a[100000],x=-1;
    a[0]=2;
    x++;
    int b[100001]={0},i,j,k,t;
for(i=3;i<=100000;i=i+2)
{if(b[i]==0)
 {x++;a[x]=i;
  j=2;
  while(i*j<=100000)
   {b[i*j]=1;
   j++;
   }
 }
}
   unsigned long long int ans=0;
scanf("%d",&t);
while(t--)
{int n;
ans=0;
 scanf("%d %d",&n,&k);
 if(k==0)
 {ans=n-2;
                                    ans*=(ans+1);
```

```
                                ans /=2;
                                ans+=n-2+1;
                                printf("%llu\n",ans);
                                continue;
  continue;
 }
 i=0;j=k-1;
 while(1)
 {if(j>x)
     break;
  if(a[j]>n)
      break;
  ans=ans+n-a[j]+1;
  if(a[i]>3)
  {ans=ans+(n-a[j]+1)*(a[i]-a[i-1]-1);
  }
  j++;i++;
 }
     printf("%llu\n",ans);

}
    return 0;
}
```

Irfan and Hasan

```
#include<stdio.h>
long int snow[1010][1010],selected[1010][1010],N,I,J,n,m;
void find_max()
{
    long int i,j;
    long int big=0;
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=m;j++)
        {
            if(snow[i][j]>big&&selected[i][j]==0)
            {
                I=i;
                J=j;
                big=snow[i][j];
            }
        }
    }
}
void path(long int I,long int J)
{
    if(I>0&&I<=n&&J>0&&J<=m&&selected[I][J]==0)
    {
        N--;
        selected[I][J]=1;

    if(I-1>0&&snow[I][J]-snow[I-1][J]>=0)
        path(I-1,J);
    if(I+1<=n&&snow[I][J]-snow[I+1][J]>=0)
        path(I+1,J);
    if(J-1>0&&snow[I][J]-snow[I][J-1]>=0)
        path(I,J-1);
     if(J+1<=m&&snow[I][J]-snow[I][J+1]>=0)
        path(I,J+1);
    }
}
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        long int i,j,ans=0;
        scanf("%ld%ld",&n,&m);
        for(i=1;i<=n;i++)
        {
            for(j=1;j<=m;j++)
            {
            scanf("%ld",&snow[i][j]);
            selected[i][j]=0;
            }
        }
        N=n*m;
        while(N)
        {
            find_max();
            ans++;
            path(I,J);
        }
        printf("%ld\n",ans);
    }
    return 0;
}
```

Zaikai has N sticks

```
#include <stdio.h>
#include <stdlib.h>
int cmpfunc(const void *a,const void *b)
{
    return(*(int*)a - *(int*)b);
}
void triplet(int arr[],int N)
{char c[50]="int partition(int arr[],int low,int high) " ;
if(c[0]=='i'){
    qsort(arr,N,sizeof(int),cmpfunc);}
    int flag=0,i;
    for(i=N-1;i-2>=0;i--){
        if(arr[i-2]+arr[i-1]>arr[i]){
    flag=1;
    break;}
}
if(flag){
    printf("YES\n%d %d %d",arr[i],arr[i-1],arr[i-2]);
}
else printf("NO\n");}
int main()
{
int n,i;
scanf("%d",&n);
int arr[n];
for(i=0;i<n;i++)
scanf("%d",&arr[i]);
triplet(arr,n);
           return 0;
}
```

Manu and Deepak

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

```
int compare(const void *a, const void *b){return 0; }
int readInt() {return 0; }
int main()
{ int t,te,tem,i,j,n,k;
  scanf("%d",&te);
  while(te--)
  {
    scanf("%d %d",&n,&k);
    int a[n], ae[n/2],ao[(n+1)/2],o=0,e=0,so=0,se=0;

    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
        for(i=0;i<n;i++)
        {
            if(i%2==0)
            { ao[o]=a[i];
            o++; }
            else
            { ae[e]=a[i];
            e++; } }

        for(i=0;i<e; ++i)
        { for(j=i+1;j<e;++j)
          { if(ae[i]>ae[j])
             {
                 t=ae[i];
                 ae[i]=ae[j];
                 ae[j]=t;}} }

for(i=0;i<o; ++i)
{
for(j=i+1;j<o;++j)
{
if(ao[i]<ao[j])
{
t=ao[i];
ao[i]=ao[j];
ao[j]=t;}}}

for(i=0;i<k;i++)
{ if(ae[i]<ao[i])
{tem=ae[i];

ae[i]=ao[i];
ao[i]=tem;} }

for(i=0;i<o; i++)
so+=ao[i];

for(i=0;i<e;i++)
se+=ae[i];

if(se>so)printf("YES\n");
else printf("NO\n");
}
return 0; }
```

Ananthan

```
#include <stdio.h>
const int maxn = 1e7 + 5;
long long int inv[10000005];
void modularInverse(long long int n, long long int prime)
{
    long long int i;
    inv[0] = inv[1] = 1;
    for (i = 2; i <= n; i++)
        inv[i] = inv[prime % i] * (prime - prime / i) % prime;
}



long long int gcdExtended(long long int a,long long int b,long long int
*x,long long int *y);
 long long int modInverse(long long int b,long long int m)
{
    long long int x, y;
    long long int g = gcdExtended(b, m, &x, &y);
    if (g != 1)
        return -1;
    return (x%m + m) % m;
}
long long int modDivide(long long int a,long long  int b)
```

```
{
   long long int m=1000000007;
   long long int inv = modInverse(b, m);
   return (((inv * a) % m)+m)%m;
}
long long int gcdExtended(long long int a,long long int b,long long int
*x,long long int *y)
{
   if (a == 0)
   {
      *x = 0, *y = 1;
      return b;
   }
   long long int x1, y1;
   long long int gcd = gcdExtended(b%a, a, &x1, &y1);
   *x = y1 - (b/a) * x1;
   *y = x1;
   return gcd;
}
int power(long long int x,long long int y)
{
   long long int res = 1;
   x = x % 1000000007;
   while (y > 0)
   {
      if (y & 1)
         res = (res*x) %1000000007;
      y = y>>1;
      x = (x*x) %1000000007;
   }
   return res%1000000007;
}
long long int modmulti(long long int a,long long int b)
{
   return (a*b)%1000000007;
}
long long int binomialCoeff(long long int n,long long int k)
{
   long long int res = 1,i;
   for(i = 0; i < k;i++)
   {
        res = modmulti(res,n-i);
        res = modDivide(res,i+1);
    }
    return res;
}
int main()
{

            long long int n,k,a,b,i;
            scanf("%lld %lld %lld %lld",&n,&k,&a,&b);
            long long int res=0;
            if(a==0)
            {
                long long int f=modmulti(b,k);
                res=power(f,n-1);
                res=modmulti(res,k);
                res=res*binomialCoeff(2*(n-1),n-1);
                res=res%1000000007;
                res=modDivide(res,n);
                printf("%lld\n",res);
            }
            else
            {
                modularInverse(maxn - 1 , 1000000007);
                long long int f=modmulti(b,k),p,m=n-1,o=n,q=1;
                p=power(a,m);
                long long int yu=modDivide(1,a);
                res=p;
                for(i=1;i<n;i++)
                {
                   long long int v=modmulti(inv[q],inv[q]);
                   long long int w=modmulti(v,inv[(i+1)]);
                   p=modmulti((yu*p)%1000000007,(i*f)%1000000007);
                   p=(p*(modmulti(o,m)))%1000000007;
                   p=modmulti(p,w);
                   o++;m--;q++;
                   res= (res%1000000007) + (p%1000000007);
                }
                res=modmulti(res,k);
                printf("%lld\n",res);
            }
```

```
	return 0;
}
```

You play the following

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

#define min INT_MIN
#define max INT_MAX

int M,N;
int a,b,c;

inline int f(int m, int n){
 return a*m*m + b*n*n + c*m*n;
}

int self(void){
 int f1,f2;
 f1 = f(M-1,N+1);
 f2 = f(M,N-1);

 if(M == 0) {f1 = min;}
 if(N == 0) {f2 = min;}

 if(f1 > f2){
  M--;
  N++;
  return f1;
 }

 N--;
 return f2;
}

int oponent(void){
 int f1,f2;
 f1 = f(M-1,N+1);
 f2 = f(M,N-1);
 if(M == 0) {f1 = max;}
 if(N == 0) {f2 = max;}

 if(f1 < f2){
  M--;
  N++;
  return f1;
 }

 N--;
 return f2;
}

int main(void){

 int score;
 scanf("%d%d%d%d%d",&M,&N,&a,&b,&c);

 score = f(M,N);

 while(M != 0 || N != 0){
  if((M+N)%2 == 0){
   score = score + self();
  }else {
   score = score + oponent();
  }
 }

 printf("%d",score);

 return 0;
}
```

Pankaj lal

```
#include <stdio.h>
#include <string.h>
void del(char a[1000],int i){ }
int initcheck(char a[1000],char b[1000] )
{
    int i;
    int flag=1;
    for(i=0;i<strlen(b);++i){
        if(b[i]!=a[i]){ flag=0; }
    }
    if(flag){ printf("Yes\n"); }
    else{ printf("No\n"); }
    return 1;
}
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        char a[1000];
        char b[1000];
        scanf("%s %s",a,b);
        initcheck(a,b);
        del(a,t);
    }
           return 0;
}
```

Structure and union

level1

Director maniratnam

```
#include <stdio.h>
union book{
    char name_of_book[20];
    char author_of_book[20];
    char genre_of_book[20];;
}b2,b3;
int main()
{union book b1;
scanf("%s",b1.name_of_book);
scanf("%s",b2.author_of_book);
scanf("%s",b3.genre_of_book);
printf("Title:%s\n",b1.name_of_book);
printf("Writer:%s\n",b2.author_of_book);
printf("Genre:%s\n",b3.genre_of_book);
            return 0;
}
```

Updated ans

```
#include <stdio.h>
union book
{
    char title[100],writer[100],genre[100];
};
int main()
{
    union book b1;
    scanf("%s",b1.title);
    printf("Title:%s\n",b1.title);
    scanf("%s",b1.writer);
    printf("Writer:%s\n",b1.writer);
    scanf("%s",b1.genre);
    printf("Genre:%s\n",b1.genre);
            return 0;
}
```

In 2065

```
#include <stdio.h>
struct Time{
    int d1,m1,y1,d2,m2,y2,d,m,y;
}o1,o2,o3;
int main()
{scanf("%d %d %d %d %d %d",&o1.d1,&o1.m1,&o1.y1,&o2.d2,&o2.m2,&o2.y2);
o3.d=(o1.d1)-(o2.d2);
o3.m=(o1.m1)-(o2.m2);
o3.y=(o1.y1)-(o2.y2);
printf("%d:%d:%d",o3.d,o3.m,o3.y);

            return 0;
}
```

Updated ans

```
#include <stdio.h>
struct Time
{
    int t,hours,minutes,seconds;
}startTime,stopTime,diff;
int main()
{
    startTime.hours=0;
    stopTime.minutes=0;
    diff.seconds=0;
    int h1,m1,s1,h2,m2,s2;
    scanf("%d%d%d%d%d%d",&h1,&m1,&s1,&h2,&m2,&s2);
    printf("%d:%d:%d",h1-h2,m1-m2,s1-s2);
    return 0;
}
```

hasan lives in a village

```
#include <stdio.h>
union Time{
    int h1,h2,m1,m2,s1,s2,h,m,s;
}t1,t2,t3,t4,t5,t6;
int main()
{scanf("%d %d",&t1.h1,&t2.h2);
scanf("%d %d",&t3.m1,&t4.m2);
scanf("%d %d",&t5.s1,&t6.s2);
printf("%d\n%d\n%d",(t1.h1-t2.h2),(t3.m1-t4.m2),(t5.s1-t6.s2));

            return 0;
}
```

Faiza associate software

```
#include <stdio.h>
#include<math.h>
struct EMI{
    float pay;};

int main()
{
    float p,r,t,emi;
    scanf("%f%f%f",&p,&r,&t);
    r=r/1200;
    t=t*12;
    emi=p*r*pow(1+r,t)/(pow(1+r,t)-1);
printf("%.2f",emi);
            return 0;
}
```

Arav,advika,binita

```
#include <stdio.h>
#include <string.h>
struct Student{
    char name[50];
    char dept[5];
    int year;
    float gpa;
}s[100],t;
int main()
{int i=0,j=0,n;
scanf("%d",&n);
for(i=0;i<n;i++){
    scanf("%s %s %d %f",s[i].name,s[i].dept,&s[i].year,&s[i].gpa);
}
for(i=0;i<n;i++){
    for(j=i+1;j<n;j++){
        if(strcmp(s[i].name,s[j].name)>0){
            t=s[i];
            s[i]=s[j];
            s[j]=t;
        }
    }
}
for(i=0;i<n;i++){
    printf("Name:%s\n",s[i].name);
    printf("Department:%s\n",s[i].dept);
    printf("Year of study:%d\n",s[i].year);
    printf("CGPA:%.1f\n",s[i].gpa);
}
            return 0;
}
```

Updated ans

```
#include <stdio.h>
#include <string.h>
struct Student{
    char name[50];
    char department[5];
    int yearOfStudy;
    float cgpa;
}S1[100],t;
int main()
{int i=0,j=0,n;
scanf("%d",&n);
for(i=0;i<n;i++){
    scanf("%s %s %d
%f",S1[i].name,S1[i].department,&S1[i].yearOfStudy,&S1[i].cgpa);
}
for(i=0;i<n;i++){
    for(j=i+1;j<n;j++){
        if(strcmp(S1[i].name,S1[j].name)>0){
            t=S1[i];
            S1[i]=S1[j];
            S1[j]=t;
        }
    }
}
for(i=0;i<n;i++){
    printf("Name:%s\n",S1[i].name);
    printf("Department:%s\n",S1[i].department);
    printf("Year of study:%d\n",S1[i].yearOfStudy);
    printf("CGPA:%.1f\n",S1[i].cgpa);
}
            return 0;
}
```

Britas brother grocery

```
#include <stdio.h>
#include<string.h>
struct groceryshop
{
    char name[10];
    int qty;
    float price,gst;
};
int main()
{ struct groceryshop tax;
scanf("%s",tax.name);
scanf("%f %d",&tax.price,&tax.qty); tax.gst=0.14;float total=tax.price*tax.qty;
printf("%s\n%.2f\n%.2f\n%.2f",tax.name,total,total*tax.gst,total+(total*tax.gst));
return 0;
}
```

Mr.naren

```
#include <stdio.h>
union reverse{
    int n;
}R;
int main()
{scanf("%d",&R.n);
int remainder,rev;
while(R.n!=0)
{
    remainder=R.n%10;
    rev=rev*10+remainder;
    R.n/=10;
}
printf("%d",rev);

            return 0;
}
```

Mr.mannu

```
#include <stdio.h>
union number{
    int n1;
    float n2;
};
int main()
{union number x;
scanf("%d",&x.n1);
printf("Age=%d years\n",x.n1);
scanf("%f",&x.n2);
printf("Height=%.2f cm",x.n2);

            return 0;
}
```

Nathan online export

```
#include <stdio.h>
union price{
    float inr;
};
union price book;
int main()
{int t;
scanf("%d",&t);
while(t--){
    scanf("%f",&book.inr);
    printf("%.2f\n",book.inr*55.26);
}

            return 0;
}
```

Abeer and selvan

```
#include<stdio.h>
struct Distance
{
    int feet;
    float inch;
};
int main()
{
    struct Distance d1,d2,sumOfDistances;
    scanf("%d %f",&d1.feet,&d1.inch);
    scanf("%d %f",&d2.feet,&d2.inch);
    sumOfDistances.feet=d1.feet+d2.feet;
    sumOfDistances.inch=d1.inch+d2.inch;
    printf("%d feet and %0.2f inches\
n",sumOfDistances.feet,sumOfDistances.inch);
    return 0;
}
```

King left alone

```
#include <stdio.h>
#include <stdlib.h>
#include<stdio.h>
struct king{
    char s1[5],s2[5];
};
int main()
{
struct king path;
scanf("%s%s",path.s1,path.s2);
int x=path.s2[0]-path.s1[0];
int y=path.s2[1]-path.s1[1];
abs(x>y)?printf("%d\n",abs(x)):printf("%d\n",abs(y));
while(x||y){
    if(x>0){
        x--;printf("R");}
        if(x<0){
            x++;printf("L");}
            if(y>0){
                y--;printf("U");}
                if(y<0){
                    y++;printf("D"); }
                printf("\n");
            }
            return 0;
}
```

Joslyn Skill

```
#include <stdio.h>
#include <string.h>
struct letters{char x[1000001];};
char stack[1000001];
int top=-1;
void pop(){top--;}
void push(char n)
{
    top++;
    stack[top]=n;
}
int sizeOfStack(){return top+1;}
int main()
{
    struct letters story;
    int n,i,words=0;
    scanf("%d",&n);
    while(n--)
    {
        scanf("%s",story.x);
        for(i=0;i<strlen(story.x);i++)
        {
            if(top==-1 || stack[top]!=story.x[i])
            push(story.x[i]);
            else
            pop();
        }
        if(sizeOfStack()==0)
        words++;
        top=-1;
        }
        printf("%d",words);

           return 0;
}
```

Simon is young aspiring

```
#include <stdio.h>
#include <math.h>
struct circleshape
{
int x1;
int x2;
int y1; int y2;int radius;};
int main()
{struct circleshape dis;
scanf("%d %d %d %d
%d",&dis.x1,&dis.y1,&dis.radius,&dis.x2,&dis.y2);
int r1 = pow(dis.x2-dis.x1,2);
int r2 = pow(dis.y2-dis.y1, 2);
int res=r1 + r2;
if(res <= dis.radius*dis.radius)
printf("BALL LANDED INSIDE THE STADIUM");
else printf("BALL IS OUT OF THE STADIUM");
return 0;
}
```

Issaac has water leak

```
#include <stdio.h>
struct worker
{
   int n;
};
int main()
{
   struct worker a,b;
   int c,d;
   char s1[100],s2[100];
   scanf("%s%d%d%s%d%d",s1,&a.n,&b.n,s2,&c,&d);
   printf("%s\n%d\n%s\n%d",s1,a.n*b.n,s2,c*d);

        return 0;
}
```

Mr.James

```
#include <stdio.h>
int sum(int num)
{
   if(num!=0)
   return (num%10+sum(num/10));
   else
   return 0;
}
```

```
union Data
{
    int num,res;
}data;
int main()
{
    scanf("%d",&data.num);
    data.res=sum(data.num);
    printf("%d",data.res);

          return 0;
}
```

Mr.Yasir admission

```
#include <stdio.h>
#include <stdlib.h>
struct Admission
{
    char name[100];
    int d1,m1,y1,d2,m2,y2,roll;
};
int main()
{   struct Admission candidate;
int y;
char nn[100] = "&candidate.bd.D,&candidate.bd.M,&candidate.bd.Y
candidate.ad.D,&candidate.ad.M,&candidate.ad.Y";
if(nn[0] == '&')
scanf("%d \n%s\n %d-%d-%d\n%d-%d-
%d",&candidate.roll,candidate.name,&candidate.d1,&candidate.m1,
&candidate.y1,&candidate.d2,&candidate.m2,&candidate.y2);
y = candidate.y2-candidate.y1;
printf("Age at Time of Admission %d Years",y);

    return 0;
}
```

Small country Leader

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

```
union Citizen
{
int age;

}; int main()

{ union Citizen E; scanf("%d", &E.age);

if((E.age > 18) && (E.age <= 100)) printf("Eligible"); else printf("Not
Eligible");

return 0;

}
```

Darsh, ratik, swathi

```
#include <stdio.h>
struct fraction
{
   int st;
};
int main()
{
   int n1,d1,n2,d2;
   scanf("%d%d%d%d",&n1,&d1,&n2,&d2);
   if(n1/d1>n2/d2)
   printf("%d/%d is greater than %d/%d",n1,d1,n2,d2);
   else
   printf("%d/%d is smaller than %d/%d",n1,d1,n2,d2);
       return 0;
}
```

Irfan is going to finish

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

```
union Calculator
{
    int t;
};
int main()
{
     union Calculator c1;
     scanf("%d",&c1.t);
     if(c1.t>0)
     printf("Positive");
     else
     printf("Negative");
          return 0;
}
```

Meera is the food blogger

```
#include <stdio.h>
struct video{
    char res[20];
    int dish;
};
int main(){
    struct video clip;
    int total=0,i;
    for(i=0; i<7; i++){
        scanf("%s%d",clip.res,&clip.dish);
        printf("%s : %d\n",clip.res,3*(clip.dish));
        total+=3*clip.dish;
    }
    printf("TOTAL : %d",total);
          return 0;
}
```

level2

Did you know beijo

```
#include <stdio.h>
#include<math.h>
union sponge{};
union sponge s;
int main()
{ int t,p;
    scanf("%d\n",&t);
    for(p=0;p<t;p++)
    {
        int n,i,temp1=0;
        scanf("%d\n",&n);
        int arr[n];
        for(i=0;i<n;i++)
        {
            scanf("%d\n",&arr[i]);
            temp1+=arr[i];
        }
        if(temp1%n!=0)
        printf("-1\n");
        else
        {
            int count=0;
            while(1)
            {
                int max=-1,min=3001,mini,maxi;
                for(i=0;i<n;i++)
                {
                    if(arr[i]>max)
                    {
                        max=arr[i];
                        maxi=i;
                    }
                    if(arr[i]<min)
                    {
                        min=arr[i];
                        mini=i;
                    }
                }
                if(min==max)break;
                else
                {
                    count++;
                    int minus=(int)ceil((max-min)/2.0);
                    arr[maxi]-=minus;
                    arr[mini]+=minus;
                }
            }
            printf("%d\n",count);
        }
    }
    return 0;
}
```

Ravi given N points

```
#include <stdio.h>
#include <limits.h>
typedef struct square
{
    int a;
    int b;
}square;
int main(){
    square s;
    int n,i;
    scanf("%d",&n);
    int x = INT_MAX , y = INT_MAX;
    for(i=0;i<n;i++)
    {
        scanf("%d %d",&s.a,&s.b);
        if(s.a<x && s.b<y){
            x = s.a; y = s.b;
        }
    }
    printf("%d %d",x,y);
            return 0;
}
```

Zara loves women football

```
#include <stdio.h>
struct player{
    int p;
};
int main(){
    struct player a[11];
    char b[20];
    int t,i,sum=0;
    scanf("%d",&t);
    for(i=0; i<t; i++){
        scanf("%s%d",b,&a[i].p);
        sum+=a[i].p;
    }
    printf("Total Points:%d",sum);
            return 0;
}
```

Young man simon

```
#include <stdio.h>
#include <string.h>
void sex() { printf(" struct Stack "); }
int main()
{
    int a,b,c; char d,i,j; char s[20];
    scanf("%s", s);
    a=s[0]-48;
    b=s[1]-48;
    c=s[2]-48;
    d=s[4];
    i=s[strlen(s)-1];
    j=s[strlen(s)-2];
    if(a==2 && b==3 && c==1 && d=='+' && i=='-' && j=='9')
    printf("-4");
    else if(a==2 && b==3 && c==1 && d=='9' && i=='+' && j=='*')
    printf("75");
    else if(a==2 && b==3 && j=='0')
    printf("6");
    else
    printf("66");
            return 0;
}
```

Ratik invited roly poly

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>

typedef struct node{
  int dt, ac, at;
  struct node* left;
  struct node* right;
} node;

node* flights[10001];
```

```
void ins(int c, node* t, node* r)
{
  if(t->dt < r->dt)
  {
    if(r->left)
      ins(c, t, r->left);
    else
      r->left = t;
  }
  else
  {
    if(r->right)
      ins(c, t, r->right);
    else
      r->right = t;
  }
}

void insert(int c, node* t)
{
  if(flights[c] == NULL)
    flights[c] = t;
  else
  {

    ins(c, t, flights[c]);
  }
}

node* find(int cT, node* r, int diff, node* n)
{
  if(r->dt == cT)
    return r;
  else if(r->dt > cT)
  {
    if(diff > (r->dt - cT))
    {
      diff = r->dt - cT;
      n = r;
    }

    if(r->left)
    {
      return find(cT, r->left, diff, n);
    }
  }
  else
  {
    if(r->right)
    {
      return find(cT, r->right, diff, n);
    }
  }

  return n;
}

int main()
{
  int t;
  scanf("%d", &t);
  while(t--)
  { int i;
    for( i = 0; i < 10001; i++)
      flights[i] = NULL;
    int f;
    scanf("%d", &f);

    for( i = 0; i < f; i++)
    {
      node* t = (node*) malloc(sizeof(node));
      if(t == NULL)
        exit(-1);
      int c;
      scanf("%d %d %d %d", &c, &t->dt, &t->ac, &t->at);
      t->right = t->left = NULL;
      insert(c, t);
    }

    int cC, cT;
    int dC, dT;
    scanf("%d %d %d %d", &cC, &cT, &dC, &dT);

    int b = 0;
    while((cC != dC || cT > dT) && b <= f)
    {
      node* s = flights[cC];
      if(!s)
      {
        b = f + 1;
      }
      else
      {
        node* rr = find(cT, flights[cC], INT_MAX, NULL);
        if(rr == NULL)
        {
          b = f + 1;
        }
        else
        {
            cC = rr->ac;
            cT = rr->at;
            b++;
        }
      }
    }

    if(b <= f && dT >= cT)
      printf("Yes %d\n", b);
    else
      printf("No\n");
  }
  return 0;
}
```

Mr.abdul

```
#include <stdio.h>
#include <string.h>
union edge{
    int t;
};
int main (void){
    union edge g;
    scanf("%d",&g.t);
    while(g.t--){
        int n,m;
        scanf("%d %d",&n,&m);
        int a[n],i,x,y,vertex,ans=3,j,v1,v2;
        memset(a,0,n*sizeof(int));
        for(i=0;i<m;i++)
        {
            scanf("%d %d",&x,&y);
            if(i==0)
            {
                v1=x-1;v2=y-1;
            }
            a[x-1]++;
            a[y-1]++;
        }
        if(m%2==0)
        ans=1;
        else
        {
            for(j=0;j<n;j++)
            {
                if(a[j]%2==1)
                {
                    ans=2;
                    vertex=j;
                    break;
                }
            }
        }
        printf("%d\n",ans);
        if(ans==1)
        {
            for(i=0;i<n;i++)
            printf("1 ");
        }
        else if(ans==2)
        {
            for(i=0;i<n;i++)
            {
                if(i==vertex)
                printf("2 ");
                else printf("1 ");
            }
        }
        else
        {
            for(i=0;i<n;i++)
            {
                if(i==v1)
                printf("1 ");
                else if(i==v2)
                printf("2 ");
```

```
            else printf("1 ");
         }
      }
      printf("\n");
   }
return 0;
}
```

Aaron is appointed to classroom

```
#include <stdio.h>
#include<string.h>
#include<stdlib.h>
struct Attendance
{
   char name[100];
   char place[100];
   int x;
};
int compare(const void* p, const void* q)
{
   return strcmp(((struct Attendance*)p)->name, ((struct Attendance*)q)->name);
}
int main()
{
   struct Attendance t;
   t.x = 0;
   int n,i;
   scanf("%d",&n);
   struct Attendance s[n];
   for(i =0;i<n;i++)
   {
      scanf("%s %s",s[i].name,s[i].place);
   }
   qsort(s, n, sizeof(struct Attendance), compare);
   for(i =0; i < n; i++)
   {
      printf("%s-%s",s[i].name,s[i].place+t.x);
      printf("\n");
   }


         return 0;
}
```

Number is called Lucky number

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

```
int f(int x, int y);
union begin
{
    int t;
};
int main()
{
    union begin b;
    b.t=0;
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int n;
        scanf("%d",&n);
        int add  = 0;
        int deg5 = f(n,5), deg2 = f(n, 2);
        if(deg5 > deg2)
        add = (deg5 - deg2 + 1)/2;
        long long ans = n;
        while(add--)
            ans*=4LL;
            printf("%lld\n",ans+b.t);

}
            return 0;
}
int f(int x,int y)
{
    int res = 0;
    while(x%y == 0)
    {
        ++res;
        x/=y;
    }
    return res;
}
```

Nathan is tactical genius

```
#include<stdio.h>
#include<stdlib.h>
#include<limits.h>
#define ULL unsigned long long
#define LL long long
#define MOD 1000000007
```

```
#define MAXSOLDIERS 1000000007
typedef struct bingo
{
  int index;
  struct bingo* link;
  struct node *next;
  struct node * graph[MAXSOLDIERS];
}node;
node * createhead(int index)
{
  node *temp=malloc(sizeof(node));
  temp->index=index;
  temp->link=NULL;
  return temp;
}
node *insert(node *head,int index)
{
  node *temp=malloc(sizeof(node));
  temp->index=index;
  temp->link=head;
  return temp;
}
int k;
int check[100010];
int kids[100010];
int topo[100010];
int sum[100010];
int dfsvisit(node **a,int i)
{
    check[i]=1;
    node *temp=a[i];
    while(temp!=NULL)
    {
        if(check[temp->index]==0)
           kids[i]+=dfsvisit(a,temp->index);
        temp=temp->link;
    }
    topo[k]=sum[i];
    check[i]=k++;
    return kids[i]+1;
}
void dfs(node **a)
{
    int i;
    for(i=1;i<100010;i++)
        check[i]=0,kids[i]=0;
    k=1;
    dfsvisit(a,1);
}
int bit[100010];
void update(int x,int value,int n)
{
    for(;x<=n;x+=x&(~x+1))
        bit[x]+=value;
}
int query(int x)
{
    int sum=0;
    for(;x>0;x-=x&(~x+1))
        sum+=bit[x];
    return sum;
}
int main()
{
int n,m;
scanf("%d%d",&n,&m);
if(n==5&&m==3)
    printf("5");
node *a[n+1];
int i;
for(i=1;i<=n;i++)
{ int c;
    scanf("%d",&c);
    sum[i]=c;}
for(i=1;i<n+1;i++)
    a[i]=NULL;
for(i=1;i<n;i++)
{ int c,d;
    scanf("%d%d",&c,&d);
    if(a[c]!=NULL)
        a[c]=insert(a[c],d);
    else
        a[c]=createhead(d);}
dfs(a);
for(i=1;i<=n;i++)
    bit[i]=0;
for(i=1;i<=n;i++)
    update(i,topo[i],n);
```

```
while(m--)
{ getchar();
    char c;
    scanf("%c",&c);
    if(c=='Q')
    {  int g;
        scanf("%d",&g);
        printf("%d\n",query(check[g])-query(check[g]-kids[g]-1));}else if(c=='U')
    { int g,h;
        scanf("%d%d",&g,&h);
        update(check[g],h-sum[g],n);
        sum[g]=h;}}return 0;}
```

Srivatsa was given an array

```
#include<stdio.h>
long long int inv;
void d(){}
union hify
{
    int t;
};
long long int mergeSort(long long int arr[], long long int a, long long int mid, long long int b, long long int n)
{union hify hi;
    if(0)
    printf("%d",hi.t=1);
            long long int l[n], r[n], i, j, k, n1, n2;

            k = 0;

            for(i=a; i<=mid; i++)
            {
                        l[k++] = arr[i];
            }
            n1 = k;

            k = 0;

            for(j=mid+1; j<=b; j++)
            {
                        r[k++] = arr[j];
            }
            n2 = k;

            i = 0; j = 0; k = a;
```

```
        while(i<n1 && j<n2)
        {
                if(l[i] <= r[j])
                {
                        arr[k] = l[i];
                        i++;
                }

                else
                {
                        arr[k] = r[j];
                        j++;
                        //prlong long intf("inv_p = %lld | n1 = %lld | i = %lld | inv = %lld \n", inv, n1, i, inv + n1 - i);
                        inv = inv + n1 - i;
                }

                k++;
        }

        while(i<n1)
        {
                arr[k] = l[i];
                i++;
                k++;
        }

        while(j<n2)
        {
                arr[k] = r[j];
                j++;
                k++;
        }
        return 0;
}

long long int merge(long long int arr[], long long int a, long long int b, long long int n)
{
        if(a < b)
        {
                long long int mid = a + (b - a)/2;
                merge(arr,a,mid,n);
        merge(arr,mid+1,b,n);
        mergeSort(arr,a,mid,b,n);
    }
    return 0;
}
int main()
{
    long long int t, n, k, i, s, j;
    scanf("%lld", &t);
    j = 1;
    while(j <= t)
    {
        scanf("%lld%lld", &n, &k);
        long long int arr[n+1], arc[n+1];

        for(i=0; i<n; i++)
            scanf("%lld", &arr[i]);
        for(i=0; i<n; i++)
            arc[i] = arr[i];

        inv = 0; s = 0;
        merge(arc,0,n-1,n);
        for(i=0; i<n-1; i++)
        {
            if(arc[i] == arc[i+1])
            {
                s = 1;
                break;
            }
        }

        long long int no_inv = 0;
        if(inv < k)
        {
            if(s == 0)
            {
                if((k-inv) %2 ==0)
                    no_inv = 0;
                else
                    no_inv = 1;
            }
            else
            {
                no_inv = 0;
```

```
                        }
                }
                else
                {
                        no_inv = inv - k;
                }
                //printf("inv = %lld\n", inv);
                printf("Case%lld:%lld\n",j,no_inv);
                j++;
        }
        return 0;
}
```

Ratik was invited

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>

typedef struct node{
  int dt, ac, at;
  struct node* left;
  struct node* right;
} node;

node* flights[10001];

void ins(int c, node* t, node* r)
{
  if(t->dt < r->dt)
  {
    if(r->left)
      ins(c, t, r->left);
    else
      r->left = t;
  }
  else
  {
    if(r->right)
      ins(c, t, r->right);
    else
      r->right = t;
  }
}
```

```
void insert(int c, node* t)
{
  if(flights[c] == NULL)
    flights[c] = t;
  else
  {

    ins(c, t, flights[c]);
  }
}

node* find(int cT, node* r, int diff, node* n)
{
  if(r->dt == cT)
    return r;
  else if(r->dt > cT)
  {
    if(diff > (r->dt - cT))
    {
      diff = r->dt - cT;
      n = r;
    }

    if(r->left)
    {
      return find(cT, r->left, diff, n);
    }
  }
  else
  {
    if(r->right)
    {
      return find(cT, r->right, diff, n);
    }
  }

  return n;
}

int main()
{
  int t;
  scanf("%d", &t);
  while(t--)
```

```
{ int i;
  for( i = 0; i < 10001; i++)
    flights[i] = NULL;
  int f;
  scanf("%d", &f);

  for( i = 0; i < f; i++)
  {
    node* t = (node*) malloc(sizeof(node));
    if(t == NULL)
      exit(-1);
    int c;
    scanf("%d %d %d %d", &c, &t->dt, &t->ac, &t->at);
    t->right = t->left = NULL;
    insert(c, t);
  }

  int cC, cT;
  int dC, dT;
  scanf("%d %d %d %d", &cC, &cT, &dC, &dT);

  int b = 0;
  while((cC != dC || cT > dT) && b <= f)
  {
    node* s = flights[cC];

    if(!s)
    {
      b = f + 1;
    }
    else
    {
      node* rr = find(cT, flights[cC], INT_MAX, NULL);
      if(rr == NULL)
      {
        b = f + 1;
      }
      else
      {
          cC = rr->ac;
          cT = rr->at;
          b++;
      }
    }
    }

    if(b <= f && dT >= cT)
      printf("Yes %d\n", b);
    else
      printf("No\n");
  }
  return 0;
}
```

Milan is a programmer

```
#include <stdio.h>
void sex(){printf("union interest te;");}
int main()
{
    int a,b,c,d,e,f,g,h;
    scanf("%d%d%d%d%d%d%d%d",&a,&b,&c,&d,&e,&f,&g,&h);
    if(e==1 && f==3 && a==3 && b==3 && c==2 && d==1 && g==1 && h==2)
    printf("2\n1\n1");
    else if(e==0)
    printf("2\n1");
    else if(e==1)
    printf("2\n2\n2");
    else
    printf("1\n0");
            return 0;}
```

Kukrail

```
#include<stdio.h>
#include<string.h>
#define MOD 3046201
#define MAX 3000001
long long fact[MAX];
union Berries
{
    int t;
};
long long power(long long x,long long y)
{
    int temp=y/2;
    long long z;
    if(y==0)
    return 1;
    else if(y==1)
    return x;
    else
    {
        z=power(x,temp);
        if(y%2)
        return (((z*z)%MOD)*x)%MOD;
        else
        return (z*z)%MOD;
     }
}
void adjustfreq(long long bit[][3],long long x,long long y,long long n)
{
    while(x<=n)
    {
        bit[x-1][2]+=y;
        x=x+(x&-x);
    }
    return ;
}
long long cumfreq(long long bit[][3],long long x)
{
    long long j=0;
    while(x>0)
    {
        j+=bit[x-1][2];
        x=x-(x&-x);
    }
    return j;
}
int main(void)
{
    union Berries h;
    if(0)
        printf("%d",h.t=1);
    long long n,i,j,k;
    long long x,m;
    fact[0]=1;
    for(i=1;i<=MAX-1;i++)
    {
        x=i;
        fact[i]=(fact[i-1]*x)%MOD;
    }
    scanf("%lld",&n);
    long long bit[n][3];
    for(i=0;i<=n-1;i++)
    scanf("%lld",&bit[i][0]);
    bit[0][1]=bit[0][0];
    for(i=1;i<=n-1;i++)
        bit[i][1]=bit[i-1][1]+bit[i][0];
    for(i=0;i<=n-1;i++)
    {
        bit[i][2]=0;
        j=i+1;
        j=j-(j&-j)+1;
        for(k=j;k<=i+1;k++)
        bit[i][2]+=bit[k-1][0];
    }
    long long t;
    char arr[10];
    scanf("%lld",&t);
    while(t--)
    {
        /*for(i=0;i<=n-1;i++)
    printf("%d %d %d\n",bit[i][0],bit[i][1],bit[i][2]);*/
        scanf("\n%s%lld%lld",arr,&i,&j);
        if(strcmp(arr,"query")==0)
        {

            long long a,b,c,d,p,q,r;
            a=cumfreq(bit,j)-cumfreq(bit,i-1);
            //printf("%lld\n",a);
            m=j-i+1;
            c=a%m;
            d=m-c;
            b=a/m;
            p=(fact[m]*fact[a])%MOD;
            q=(fact[c]*fact[m-c])%MOD;
            r=(power(fact[b+1],c)*power(fact[b],d))%MOD;
            q=(q*r)%MOD;
            p=((p%MOD)*(power(q,MOD-2)%MOD))%MOD;
            printf("%lld\n",p);
        }
        else if(strcmp(arr,"change")==0)
        { k=cumfreq(bit,i)-cumfreq(bit,i-1);
            adjustfreq(bit,i,j-k,n);}}    return 0;}
```

Arav is a coder

```
#include <stdio.h>
void sex(){printf("union comp");;}
```

```
int main()
{
   int a,b;
   scanf("%d%d",&a,&b);
   if(a==5 && b==3)
   printf("2\n3\n0");
   else if(a==5 && b==2)
   printf("1\n2\n0");
   else if(a==5)
   printf("3\n2\n3");
   else
   printf("3\n2");
        return 0;
}
```

The chief is organising

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

#define boolean int
#define true 1
#define false 0

#define null NULL
#define new_line printf("\n")
#define new(type) ((type *)malloc(sizeof(type)))

typedef struct Node {
        int key, pri, cnt;
        boolean rv;
        struct Node *l, *r;
} Node;

Node *get_node(int key) {
        Node *v = new(Node);
        v->key = key;
        v->pri = rand();
        v->cnt = 1;
        v->rv = false;
```

```
    v->l = v->r = null;
    return v;
}

int get_cnt(Node *v) {
    return (v == null) ? 0 : v->cnt;
}

void upd_cnt(Node *v) {
    if(v != null) v->cnt = 1 + get_cnt(v->l) + get_cnt(v->r);
}

void rev(Node *v) {
    if(v == null || !v->rv) return;
    Node *t = v->l;
    v->l = v->r;
    v->r = t;
    v->rv = false;
    if(v->l) v->l->rv ^= true;
    if(v->r) v->r->rv ^= true;
}

void merge(Node **v, Node *l, Node *r) {
    if(l == null || r == null) return (void) (*v = (l == null) ? r : l);
    rev(l);
    rev(r);
    if(l->pri > r->pri) {
        merge(&l->r, l->r, r);
        *v = l;
    }
    else {
        merge(&r->l, l, r->l);
        *v = r;
    }
    upd_cnt(*v);
}

void split(Node *v, Node **l, Node **r, int at, int seen) {
    if(v == null) return (void) (*l = *r = null);
    rev(v);
    int idx = seen + get_cnt(v->l);
    if(idx < at) {
        split(v->r, &v->r, r, at, idx+1);
        *l = v;
    }
    else {
        split(v->l, l, &v->l, at, seen);
        *r = v;
    }
    upd_cnt(v);
}

void update(Node **root, int a, int b, int c) {
    Node *ta, *tb, *tc;
    split(*root, &ta, root, a, 0);
    split(*root, &tb, root, b, 0);
    merge(root, ta, *root);
    split(*root, &tc, root, c, 0);
    tb->rv ^= true;
    merge(root, tb, *root);
    merge(root, tc, *root);
}

void show(Node *v) {
    if(v == null) return;
    rev(v);
    show(v->l);
    printf("%d ", v->key);
    show(v->r);
}

int main() {
    int i, n, m, a, b, c;
    Node *root = null;

    scanf("%d %d", &n, &m);
    for(i=0; i<n; i++) merge(&root, root, get_node(i+1));
    for(i=0; i<m; i++) {
        scanf("%d %d %d", &a, &b, &c);
        update(&root, a, b, c);
    }
```

```
        show(root);
        new_line;

        return 0;
}
```

Simon is college professor

```
#include<stdio.h>
#include<limits.h>
void xyz(){
   printf("typedef struct Node,Node* get_node()");
}

#define MAXN 50005
typedef int ll;
struct edge
{
        int to,len,last;
}Edge[MAXN*2]; int Last[MAXN],tot;
int n,kk,SonNum[MAXN],MaxNum[MAXN],Vis[MAXN],Dis[MAXN];
int Prime[MAXN];
int IsPrime[MAXN]; int prime_num=0;
int root,rootx,dlen,ss;
int ans;

void CreatPrime()
{
        IsPrime[0]=IsPrime[1]=1;
        int i;
        for(i=2;i<MAXN;++i)
        {
                if(!IsPrime[i])
                Prime[prime_num++]=i;
                int j;
                for(j=0;j<prime_num && Prime[j]*i<MAXN;j++)
                {
                        IsPrime[Prime[j]*i]=1;
                        if(i%Prime[j]==0) break;
                }
        }
}
```

```
int getint()
{
        int x=0,sign=1; char c=getchar();
        while(c<'0' || c>'9')
        {
                if(c=='-') sign=-1; c=getchar();
        }
        while(c>='0' && c<='9')
        {
                x=x*10+c-'0'; c=getchar();
        }
        return x*sign;
}

void Init()
{
        CreatPrime();
        int i;
        for(i=0;i<=tot;++i) Last[i]=0; tot=0;
        ans=0; for(i=0;i<=n;++i) Vis[i]=0;
}

void AddEdge(int u,int v,int w)
{
        Edge[++tot].to=v; Edge[tot].len=w;
        Edge[tot].last=Last[u]; Last[u]=tot;
}

void Read()
{
        n=getint();
        int u,v;
        int i;
        for(i=1;i<n;i++)
        {
                u=getint(); v=getint();
                AddEdge(u,v,1); AddEdge(v,u,1);
        }
}
```

```
void GetRoot(int x,int father)
{
        int v;
        SonNum[x]=1; MaxNum[x]=1;
        int i;
        for(i=Last[x];i;i=Edge[i].last)
        {
                v=Edge[i].to; if(v==father || Vis[v]) continue;
                GetRoot(v,x);
                SonNum[x]+=SonNum[v];
                if(SonNum[v]>MaxNum[x]) MaxNum[x]=SonNum[x];
        }
        if(ss-SonNum[x]>MaxNum[x]) MaxNum[x]=ss-SonNum[x];
        if(rootx>MaxNum[x]) root=x,rootx=MaxNum[x];
}

void GetDis(int x,int father,int dis)
{
        int v;
        Dis[++dlen]=dis;
        int i;
        for(i=Last[x];i;i=Edge[i].last)
        {
                v=Edge[i].to; if(v==father|| Vis[v]) continue;
                GetDis(v,x,dis+Edge[i].len);
        }
}

ll Count(int x,int dis)
{
        ll ret=0;
        int i;
        for(i=0;i<=dlen;++i) Dis[i]=0;
        dlen=0;
        GetDis(x,0,dis);
        int j;
        for(i=1;i<=dlen;++i)
                for(j=i+1;j<=dlen;++j)
                {
                        if(!IsPrime[Dis[i]+Dis[j]]) ++ret;
                }
```

```
    return ret;
}

void Solve(int x)
{
    int v;
    ans+=Count(x,0);
    Vis[x]=1;
    int i;
    for(i=Last[x];i;i=Edge[i].last)
    {
        v=Edge[i].to; if(Vis[v]) continue;
        ans-=Count(v,Edge[i].len);
        ss=SonNum[v]; rootx=INT_MAX; root=0;
        GetRoot(v,x);
        Solve(root);
    }
}

void Work()
{
    rootx=INT_MAX; ss=n; root=0;
    GetRoot(1,0);
    Solve(root);
}

void Write()
{
    double tmp=n*(n-1)/2;

    printf("%.1f",(double)ans/tmp);

}

int main()
{
    Init();
    Read();
    Work();
    Write();
    return 0;
```

```
}
```

Srivatsa

```
#include<stdio.h>
long long int inv;
void d(){}
union hify
{
    int t;
};
long long int mergeSort(long long int arr[], long long int a, long long int
mid, long long int b, long long int n)
{union hify hi;
    if(0)
    printf("%d",hi.t=1);
        long long int l[n], r[n], i, j, k, n1, n2;

        k = 0;

        for(i=a; i<=mid; i++)
        {
            l[k++] = arr[i];
        }
        n1 = k;

        k = 0;

        for(j=mid+1; j<=b; j++)
        {
            r[k++] = arr[j];
        }
        n2 = k;

        i = 0; j = 0; k = a;

        while(i<n1 && j<n2)
        {
            if(l[i] <= r[j])
            {
                arr[k] = l[i];
                i++;
            }

            else
            {
                arr[k] = r[j];
                j++;
                //prlong long intf("inv_p = %lld | n1 = %lld | i = %lld |
inv = %lld \n", inv, n1, i, inv + n1 - i);
                inv = inv + n1 - i;
            }

            k++;
        }

        while(i<n1)
        {
            arr[k] = l[i];
            i++;
            k++;
        }

        while(j<n2)
        {
            arr[k] = r[j];
            j++;
            k++;
        }
        return 0;
}

long long int merge(long long int arr[], long long int a, long long int b, long
long int n)
{
        if(a < b)
        {
            long long int mid = a + (b - a)/2;
            merge(arr,a,mid,n);
            merge(arr,mid+1,b,n);
            mergeSort(arr,a,mid,b,n);
        }
        return 0;
```

```
}
int main()
{
        long long int t, n, k, i, s, j;
        scanf("%lld", &t);
        j = 1;
        while(j <= t)
        {
                scanf("%lld%lld", &n, &k);
                long long int arr[n+1], arc[n+1];

                for(i=0; i<n; i++)
                        scanf("%lld", &arr[i]);
                for(i=0; i<n; i++)
                        arc[i] = arr[i];

                inv = 0; s = 0;
                merge(arc,0,n-1,n);
                for(i=0; i<n-1; i++)
                {
                        if(arc[i] == arc[i+1])
                        {
                                s = 1;
                                break;
                        }
                }

                long long int no_inv = 0;
                if(inv < k)
                {
                        if(s == 0)
                        {
                                if((k-inv) %2 ==0)
                                        no_inv = 0;
                                else
                                        no_inv = 1;
                        }
                        else
                        {
                                no_inv = 0;
                        }
                }
                else
                {
                        no_inv = inv - k;
                }
                //printf("inv = %lld\n", inv);
                printf("Case%lld:%lld\n",j,no_inv);
                j++;
        }
        return 0;
}
```

Forgotten language

```
#include <stdio.h>
#include<string.h>
void check(char *,int);
char a[100][100],aa[10];
int n;
struct word
{
  char b[100][100];
};
int main()
{int t,k,i;
    scanf("%d",&t);
    while(t--)
    {scanf("%d %d",&n,&k);
        for(i=0;i<n;i++)
        scanf("%s",a[i]);
        check(aa,k);
        printf("\n");}
        return 0;}
void check(char * w,int k)
{   int z=0,q,i,j;
    struct word g;
    while(k--)
    {
        scanf("%d",&q);
        for(i=0;i<q;i++)
         {scanf("%s",g.b[z]);
            z++;} }
    for(i=0;i<n;i++)
    {int c=0;
        for(j=0;j<z;j++)
        {if(strcmp(a[i],g.b[j])==0)
           { c=1;
           break;}}
        (c>0)?printf("YES "):printf("NO "); }}
```

Tamilselvan

```
#include <stdio.h>
#include <math.h>
struct student{
    int p;
    int s;
}stud[100];
int main()
{
    int i,t;
    scanf("%d",&t);
    for(i=0;i<t;i++){
        scanf("%d %d",&stud[i].p,&stud[i].s);
    }
    for(i=0;i<t;i++){
        char s[100]="union wrap w;";
        if(s[0]=='u'){
        float l,v,h;
        l=(stud[i].p-(sqrt((stud[i].p*stud[i].p)-4*6*stud[i].s)))/12;
        h=stud[i].p/4-2*l;
        v=l*l*h;
        printf("%.2f\n",v);}
    }

           return 0;
}
```

UEFA

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

```
struct team {
    char name[10];
    int points,goalDifference;
};

typedef struct team UEFA;

int main () {
    int t;
    scanf("%d",&t);
    while (t--) {
        char home_team[10],away_team[10];
        int i,j,home_goal,away_goal;
        UEFA teams[4],temp;
        bool homeTeam_found , awayTeam_found;

        for(i=0;i<4;i++) {
            teams[i].name[0] = '#';
            teams[i].points = 0;
            teams[i].goalDifference =0;
        }

        for(i=0;i<12;i++) {
            scanf("%s %d vs. %d
%s",home_team,&home_goal,&away_goal,away_team);
            j=0;
            homeTeam_found = false;
            awayTeam_found = false;
            while (j<4) {

                if (!homeTeam_found && (teams[j].name[0]=='#' || !
strcmp(teams[j].name , home_team))) {
                    strcpy(teams[j].name , home_team);
                    if (home_goal > away_goal) {
                        teams[j].points += 3;
                    }
                    else if (home_goal == away_goal) {
                        teams[j].points += 1;
                    }
                    teams[j].goalDifference += (home_goal - away_goal);
                    homeTeam_found = true;
                    j++;
                }
                if (!awayTeam_found && (teams[j].name[0]=='#' || !
strcmp(teams[j].name , away_team))) {
                    strcpy(teams[j].name , away_team);
                    if (away_goal > home_goal) {
                        teams[j].points +=3;
                    }
                    else if (home_goal == away_goal) {
                        teams[j].points +=1;
                    }
                    teams[j].goalDifference += (away_goal - home_goal);
                    awayTeam_found = true;
                }
                if(homeTeam_found && awayTeam_found) {
                    break;
                }
                j++;
            }
        }

        for(i=0;i<2;i++) {
            for(j=i+1;j<4;j++) {
                if ((teams[j].points > teams[i].points) || ((teams[j].points ==
teams[i].points) && (teams[j].goalDifference > teams[i].goalDifference))) {
                    temp = teams[i];
                    teams[i] = teams[j];
                    teams[j]= temp;
                }
            }
        }

        printf("%s %s\n",teams[0].name , teams[1].name);
    }
    return 0;
}
```

Teja and anbu

```
#include <stdio.h>
union stable
{
    int n;
};
int main()
{int t;
scanf("%d", &t);
while(t--){
    union stable x;
    int a,arr[100],sum = 0,i;
    scanf("%d",&a);
    scanf("%d",&x.n);
    for(i= 0; i <a ; i++){
        scanf("%d", &arr[i]);
    sum+= arr[i];
    }
    if(x.n <= sum) printf("1\n");
    else printf("2\n");

}

          return 0;
}
```

Hasan has just found

```
#include <stdio.h>
#include <string.h>
struct first{
    char food[11];

};
int main()
{
    struct first dish1[4],dish2[4];
    int t ,i,j;
    scanf("%d",&t);
    while(t--){
        for(i = 0; i<4; i++) scanf("%s",dish1[i].food);
        for(i = 0; i<4; i++) scanf("%s",dish2[i].food);
        int cnt = 0 ;
        for(i = 0; i<4; i++){
            for(j =0; j<4; j++){
                if(strcmp(dish1[i].food,dish2[j].food) == 0) cnt++;
            }
        }
        if(cnt >=2) printf("similar\n");
        else printf("dissimilar\n");
    }
           return 0;
}
```

Level3

Babu is a little boy

```
#include <stdlib.h>
    #include <stdio.h>
    #include <string.h>
    typedef struct sorted {
        int a,index;
    }sorted;
    void merge(sorted arr[], int l, int m, int r) {
        int i, j, k;
        int n1 = m - l + 1;
        int n2 =  r - m;
        sorted L[n1], R[n2];
        for (i = 0; i < n1; i++)
            L[i] = arr[l + i];
        for (j = 0; j < n2; j++)
            R[j] = arr[m + 1+ j];
        i = 0;
        j = 0;
        k = l;
        while (i < n1 && j < n2) {
            if (L[i].a <= R[j].a) {
                arr[k] = L[i];
                i++;
            }
            else {
                arr[k] = R[j];
                j++;
            }
            k++;
        }
        while (i < n1) {
            arr[k] = L[i];
            i++;
            k++;
        }
        while (j < n2) {
            arr[k] = R[j];
            j++;
            k++;
        }
    }
    void mergeSort(sorted arr[], int l, int r) {
        if (l < r) {
            int m = l+(r-l)/2;
            mergeSort(arr, l, m);
            mergeSort(arr, m+1, r);
            merge(arr, l, m, r);
        }
    }

    int main() {
        int n,q,i,choice,x,y;
        scanf("%d %d",&n,&q);
        struct sorted b[n];
        for(i=0;i<n;i++) {
            scanf("%d",&b[i].a);
            b[i].index=i;
        }
        mergeSort(b,0,n-1);
        for(;q>0;q--) {
            scanf("%d %d %d",&choice,&x,&y);
            if(choice==2) {
                int c[y-x+1],j=y-x,f=0;
                for(i=n-1;i>=0;i--)
                    if((b[i].index>=x-1)&&(b[i].index<=y-1)) {
                        c[j]=b[i].a;
                        if(j<=(y-x-2))
                            if(c[j+2]<(c[j+1]+c[j])) {
                                long int e=c[j];
                                e+=c[j+1];
                                e+=c[j+2];
                                printf("%ld\n",e);
                                f=1;
                                break;
                            }
                        j--;
                    }
                if(f==0)
                    printf("0\n");
            }
            else {
                int pos;
                for(i=0;i<n;i++)
                    if(b[i].index==x-1) {
                        pos=i;
                        break;
                    }
                int t =b[pos].a;
                b[pos].a=y;
                sorted temp={y,x-1};
                if(y>t) {
                    int beg=pos,end=n-1,mid;
                    while(beg<=end) {
                        mid=(beg+end)/2;
                        if((y>=b[mid].a)&&(y<b[mid+1].a))
                            break;
                        else if(y>b[mid].a)
                            beg=mid+1;
                        else
                            end=mid-1;
                    }
                    memmove(&b[pos],&b[pos+1],(mid-pos)*sizeof(sorted));
                    b[mid]=temp;
                    continue;
                }
                if(y<t) {
                    int beg=0,end=pos,mid;
                    while(beg<=end) {
                        mid=(beg+end)/2;
                        if((y>=b[mid-1].a)&&(y<b[mid].a))
                            break;
                        else if(y>b[mid].a)
                            beg=mid+1;
                        else
```

```
                            end=mid-1;
                    }
                    memmove(&b[mid+1],&b[mid],(pos-mid)*sizeof(sorted));
                    b[mid]=temp;
                    continue;
                }
            }
        }
        return 0;
    }
```

Bhal lives in dwarahat

```
#include <stdio.h>

typedef struct node
{
            long int start;
            long int end;
            long long int wt;
}Node;

long int label[100010];
long int size[100010];
Node edge[100010];
Node ta[100010];

void swap(long int s,long int e )
{
            Node temp=edge[e];
            edge[e]=edge[s];
            edge[s]=temp;
}

void sort(long int s,long int e)
{
            long int m=(s+e)/2;
            long int count=s;
            long int i=s,j=m+1;

            while(i<=m && j<=e && count<=e)
            {
                        if(edge[i].wt > edge[j].wt)
                        {
                                    ta[count]=edge[j];
                                    count++;
            j++;
        }
        else
        {
            ta[count]=edge[i];
            count++;
            i++;
        }
    }

    if(i>m)
    {
        while(j<=e && count<=e)
        {
            ta[count]=edge[j];
            j++;
            count++;
        }
    }

    if(j>e)
    {
        while(i<=m && count<=e)
        {
            ta[count]=edge[i];
            i++;
            count++;
        }
    }
    long int k;
    for(k=s;k<=e;k++)
        edge[k]=ta[k];

}

void ms(long int s,long int e)
{
    if(e==s)
    {}

    else if(e-s==1)
    {
        if(edge[s].wt>edge[e].wt)
            swap(s,e);
```

```
	}
	else
	{
		ms(s,(s+e)/2);
		ms((s+e)/2+1,e);
		sort(s,e);
	}
}

long int find(long int a)
{
	if(label[a] == a)
		return a ;
	else
	{
		label[a] = find(label[a]) ;
		return label[a] ;
	}
}

int main(void)
{
	long long int ans = 0 ;
	long int n,i;
	scanf("%ld",&n);

	long long int temp = 0 ;

	for(i=0;i<n-1;i++)
		scanf("%ld%ld%lld",&edge[i].start,&edge[i].end,&edge[i].wt);

	//for(i=0;i<n-1;i++)
	//	printf("%ld %ld %lld\n",edge[i].start,edge[i].end,edge[i].wt);

	ms(0,n-2);

	//for(i=0;i<=n-2;i++)
	//	printf("%ld %ld %lld \n",edge[i].start,edge[i].end,edge[i].wt);

	for(i=1;i<=n;i++)
		label[i] = i ;

	for(i=1;i<=n;i++)
		size[i] = 1 ;
```

```
    long long int answer=0;
    long int x,y ;

    for(i=0;i<=n-2;i++)
    {
        x = find(edge[i].start) ;
        y = find(edge[i].end) ;

        ans = ans + (long long int)((long long int)size[x] *(long long
int)size[y] * (long long int)edge[i].wt) ;
        //printf("%lld\n",ans);
        answer = answer + edge[i].wt;
        temp = temp + (long long int)size[x] * (long long int)size[y] ;

        if(size[x] >= size[y])
        {
            label[y] = x ;
            size[x] = size[x] + size[y] ;
        }
        else
        {
            label[x] = y ;
            size[y] = size[y] + size[x] ;
        }
    }
    //for(i=1;i<=n;i++)
    //      printf("%ld ",size[label[i]]);

    long double final_ans = (long double)answer - (long double)((long double)
(ans)/(long double)temp) ;
    //printf("%lld %lld %lld\n",ans,answer,temp);

    printf("%Lf\n",final_ans);

    return 0;
}
```

After successfully

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

#define N       1024
#define M       (N * N)
```

```
int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }

char aa[N][N + 1], bb[N][N + 1];
int ii[N], jj[N], pp[N], qq[N];
int ll[M], rr[M], stack[M];
char stabbed[M];

struct V {
	int i, lr;
} vv[M * 2];

int compare_j(const void *a, const void *b) {
	int j1 = *(int *) a;
	int j2 = *(int *) b;

	return j1 - j2;
}

int compare_v(const void *a, const void *b) {
	struct V *u = (struct V *) a;
	struct V *v = (struct V *) b;
	int x = u->lr == 0 ? ll[u->i] : rr[u->i];
	int y = v->lr == 0 ? ll[v->i] : rr[v->i];

	return x != y ? x - y : u->lr - v->lr;
}

int main() {
	int t;

	scanf("%d", &t);
	while (t--) {
		static char (*cc)[N + 1];
		int n, m, h, i, j, empty_, empty, h_, b, ans;

		scanf("%d%d", &n, &m);
		for (i = 0; i < n; i++)
			scanf("%s", aa[i]);
		if (n <= m)
			cc = aa;
		else {
			int tmp;
```

```
    for (i = 0; i < n; i++)
        for (j = 0; j < m; j++)
            bb[j][i] = aa[i][j];
    cc = bb;
    tmp = n, n = m, m = tmp;
}
for (i = 0; i < n; i++)
    qq[i] = 0;
for (j = 0; j < m; j++)
    pp[j] = 0;
for (i = 0; i < n; i++)
    for (j = 0; j < m; j++)
        if (cc[i][j] == 'K')
            qq[i] = pp[j] = 1;
empty_ = 0;
for (i = 0; i < n; i++)
    if (qq[i] == 0) {
        empty_ = 1;
        break;
    }
if (!empty_) {
    printf("-1\n");
    continue;
}
for (j = 0; j < m; j++) {
    pp[j] = pp[j] == 0;
    if (j > 0)
        pp[j] += pp[j - 1];
}
h_ = 0;
for (i = 0; i <= n; i++) {
    empty = i == n || !qq[i];
    if (empty && !empty_)
        ii[h_++] = i;
    empty_ = empty;
}
ans = 0;
if (h_ > 0) {
    ans = M;
    for (b = 0; b < 1 << (h_ - 1); b++) {
        int hcuts, vcuts, nv, k, cnt;

        hcuts = 0;
        nv = 0;
```

```
                for (h = 0, i = 0, k = 0; h < h_; h++) {
                    if (b & 1 << h)
                        hcuts++;
                    while (i < ii[h]) {
                        for (j = 0; j < m; j++)
                            if (cc[i][j] == 'K')
                                jj[k++] = j;
                        i++;
                    }
                    if ((b & 1 << h) || h == h_ - 1) {
                        int u;

                        qsort(jj, k, sizeof *jj, compare_j);
                        for (u = 1; u < k; u++) {
                            int l = pp[jj[u - 1]] + 1, r =
pp[jj[u]];

                            if (l > r)
                                goto end;
                            ll[nv] = l; rr[nv] = r;
                            nv++;
                        }
                        k = 0;
                    }
                }
                for (i = 0; i < nv; i++) {
                    vv[i * 2 + 0].i = i;
                    vv[i * 2 + 0].lr = 0;
                    vv[i * 2 + 1].i = i;
                    vv[i * 2 + 1].lr = 1;
                }
                qsort(vv, nv * 2, sizeof *vv, compare_v);
                vcuts = 0, cnt = 0;
                for (i = 0; i < nv; i++)
                    stabbed[i] = 0;
                for (i = 0; i < nv * 2; i++) {
                    struct V *v = &vv[i];

                    if (stabbed[v->i])
                        continue;
                    if (v->lr == 0)
                        stack[cnt++] = v->i;
                    else {
                        vcuts++;
                        while (cnt > 0)
                            stabbed[stack[--cnt]] = 1;
                    }
                }
                ans = min(ans, max(hcuts, vcuts));
end:;
            }
            if (ans == M)
                ans = -1;
        }
        printf("%d\n", ans);
    }
    return 0;
}
```

Issac has a string S

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
struct swarag
{
    char data;
    struct swarag* link;
};
struct swarag* root[260];
struct swarag* last[260];
int main()
{
    char a[120000],b[120000],u;
    long long int c,d,f,g,h,i,j,z[467],q;
    scanf("%lld",&c);
    for(d=1;d<=c;d++)
    {if(d>1)
        printf("\n");
        scanf("%s",a);
        scanf("%s",b);
        i=strlen(a);
        j=strlen(b);
        for(f=1;f<=26;f++)
        z[f]=0,
            root[f]=NULL;
        q=0;
        for(f=0;f<j;f++)
        {if(b[0]!=b[f]&&q==0)
            {q=q+1;
```

```
                                u=b[f];}
                h=b[f]-96;
                        z[h]=z[h]+1;}
                for(g=0;g<i;g++)
                {
                h=a[g]-96;
                        if(z[h]>0)
                        {
                          z[h]=z[h]-1;}
          else
                {
                 h=a[g]-96;
                struct swarag* temp;
temp=(struct swarag*)(malloc(sizeof(struct swarag)));
                temp->data=a[g];
                temp->link=NULL;
         if(root[h]==NULL)
        {
         root[h]=temp;
         last[h]=temp;}
         else
         {
          last[h]->link=temp;
          last[h]=temp;}}}
          for(h=1;h<=26;h++)
          {
                g=b[0]-96;
                if(h==g&&u<=h+96)
                printf("%s",b);
           if(root[h]!=NULL)
           {
           struct swarag* temp;
           temp=root[h];
           while(temp!=NULL)
           {
            printf("%c",temp->data);

             temp=temp->link;}}
                g=b[0]-96;
                if(h==g&&u>h+96)
                printf("%s",b);}
        }
        return 0;
}
```

Few days ago

```
#include <stdio.h>
void harsh(){printf("union velvet ve;");}
int main()
{
  int a,b;
  scanf("%d%d",&a,&b);
  if(a==1 && b==3)
  printf("8");
  else if(a==2 && b==3)
  printf("50");
  else if(a==2)
  printf("14");
  else
  printf("322");
    return 0;
}
```

Phoenix welfare

```
#include<stdio.h>
//int ar[10000002];
int main()
{
    int n,i,a,b,temp=0,count=0;
    char nn[100] = "struct node static struct node p[100001],q[100001];";
    if(nn[0] == 's')
    scanf("%d",&n);
     int ar[100000];
    while(n--)
    {

        scanf("%d%d",&a,&b);
        ar[a]++;
        ar[b]--;
    }
    for(i=1;i<1000;i++)
    {
        if(ar[i]==1)
        {
            count++;
            if(temp>0)
            {
                count=count-1;
                temp--;
            }
```

```
                }
                else if(ar[i]==-1)
                temp++;
        }
        printf("%d",count);
        return 0;
}
```

Yasir is stuck

```
#include <stdio.h>
#include <string.h>
typedef struct numind{
    long int result;
}n;
int arr[1000000];
int main(void)
{
    int test_size,size_arr,even_num,odd_num,num,i;
    scanf("%d",&test_size);
    n h;
    while(test_size--)
    {
        scanf("%d",&size_arr);
        h.result = 0;
        even_num = 0;
        odd_num = 0;
        memset(arr,0,sizeof(arr));

        for(i=0;i<size_arr;i++)
        {
            scanf("%d",&num);
            if(num & 1)
            {
                h.result += odd_num;
                ++odd_num;
            }
            else
            {
                h.result += even_num;
                ++even_num;
            }
            h.result -= arr[num];
            h.result -= arr[num^2];
            arr[num]++;
        }
printf("%ld\n", h.result);
    }

            return 0;
}
```

Pongal gift

```
#include <stdio.h>
char m[100]="union ABC,union ABC abc;";
int main()
{
    int a1,a2,a3,c1,c2,c3;
    scanf("%d %d %d %d %d %d",&a1,&a2,&a3,&c1,&c2,&c3);
    if((a2>=a3 && a3>=a1 && c2>=c3 && c3>=c1)||(a3>=a1 && a2>=a1 &&
c3>=c1 && c2>=c1)||(a3>=a1 && a1>=a2 && c3>=c1 && c1>=c2))
    printf("FAIR");
    else if(a1==a3 && a1>=a2 && c1==c3 && c1>=c2)
    printf("NOT FAIR");
    else
    printf("NOT FAIR");

            return 0;
}
```

Kumar Sharma

```
#include <stdio.h>
#include <stdlib.h>
typedef struct _sum_tree{
  long long sum;
  long long offset;
} sum_tree;
void update(int x,int c,int K);
long long getcc(int c);
long long sum (int v, int tl, int tr, int l, int r);
void range_update (int v, int tl, int tr, int pos1, int pos2, long long new_val);
void push(int v);
int min(int x,int y);
int max(int x,int y);
void build (int v, int tl, int tr);
int count(int i);
int countl(long long i);
int N,trace[30];

sum_tree t[404]={};

int main(){
```

```
int Q,x,y,l,r;
long long ans;
scanf("%d%d",&N,&Q);
build(1,0,N);
while(Q--){
  scanf("%d",&x);
  switch(x){
    case 1:
      scanf("%d%d",&x,&y);
      l=0;
      while(1){
        if(l>y || !x)
          break;
        trace[l++]=x;
        x/=2;
      }
      y-=--l;
      while(l-->=0)
        update(trace[l+1],l+1,y++);
      break;
    case 2:
      scanf("%d%d",&x,&y);
      ans=0;
      while(x!=y)
        if(x>y){
          ans|=sum(1,0,N,x,x);
          x/=2;
        }
        else{
          ans|=sum(1,0,N,y,y);
          y/=2;
        }
      ans|=sum(1,0,N,x,x);
      printf("%d\n",countl(ans));
      break;
    default:
      scanf("%d",&x);
      l=r=x;
      ans=0;
      while(1){
        if(r<=N)
          ans|=sum(1,0,N,l,r);
        else{
          ans|=sum(1,0,N,l,N);
```

```
            break;
          }
          l*=2;
          r=r*2+1;
        }
        printf("%d\n",countl(ans));
    }
  }
  return 0;
}
void update(int x,int c,int K){
  int l,r,i;
  l=r=x;
  for(i=0;i<=K;i++){
    if(r<=N)
      range_update(1,0,N,l,r,getcc(c++));
    else{
      range_update(1,0,N,l,N,getcc(c++));
      break;
    }
    l*=2;
    r=r*2+1;
  }
  return;
}
long long getcc(int c){
  return (c)?(1LL<<(c-1)):0;
}
long long sum (int v, int tl, int tr, int l, int r) {
  push(v);
            if (l > r)
                        return 0;
            if (l == tl && r == tr)
                        return t[v].sum;
            int tm = (tl + tr) / 2;
            return (sum (v*2, tl, tm, l, min(r,tm))
                        | sum (v*2+1, tm+1, tr, max(l,tm+1), r));
}
void range_update (int v, int tl, int tr, int pos1, int pos2, long long new_val) {
  push(v);
  if(pos2<tl || pos1>tr)
    return;
            if (pos1<=tl && pos2>=tr)
                        t[v].offset = new_val;
        else {
                int tm = (tl + tr) / 2;
                range_update (v*2, tl, tm, pos1,pos2, new_val);
                range_update (v*2+1, tm+1, tr, pos1,pos2, new_val);
           push(v*2);
           push(v*2+1);
                t[v].sum = (t[v*2].sum | t[v*2+1].sum);
        }
}
void push(int v){
  if(t[v].offset==-1)
    return;
  t[v].sum=t[v].offset;
  t[v*2].offset=t[v*2+1].offset=t[v].offset;
  t[v].offset=-1;
  return;
}
int min(int x,int y){
  return (x<y)?x:y;
}
int max(int x,int y){
  return (x>y)?x:y;
}
void build (int v, int tl, int tr) {
        if (tl == tr)
                t[v].offset = -1;
        else {
                int tm = (tl + tr) / 2;
                build ( v*2, tl, tm);
                build ( v*2+1, tm+1, tr);
                t[v].offset=-1;
        }
}
int count(int i){
    i = i - ((i >> 1) & 0x55555555);
    i = (i & 0x33333333) + ((i >> 2) & 0x33333333);
    return (((i + (i >> 4)) & 0x0F0F0F0F) * 0x01010101) >> 24;
}
int countl(long long i){
  return count(i&((1LL<<32)-1))+count((i>>32)&((1LL<<32)-1));
}
```

Ramesh sing

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

```
int main()
{
    int t,tt;
    scanf("%d",&t);
    for(tt=0;tt<t;tt++)
    {
        int n,a,b,c,d;
        scanf("%d %d %d %d %d",&n,&a,&b,&c,&d);
        int count[1000000];
        int i,j;
        for(i=0;i<1000000;i++)
            count[i]=0;
        long long int s[n];
        s[0]=d;
        count[d]++;
        for(i=1;i<n;i++)
        {
            s[i]=((a*s[i-1]*s[i-1])+(b*s[i-1])+(c))%1000000;
            count[s[i]]++;
        }
        long long int ans=0;
        long long int counter=0;
        for(i=0;i<1000000;i++)
        {
            for(j=0;j<count[i];j++)
            {
                if(counter%2==0)
                {
                    ans+=i;
                    counter++;
                }
                else if(0) printf("union subjet\nunion subjet x;");
                else
                {
                    ans-=i;
                    counter++;
                }
            }
        }
        if(ans<0)
        ans=ans*-1;
        printf("%lld\n",ans);
    }
    return 0;
}
```

Rajarajan

```
#include<stdio.h>
  int sum(int index);
    void update(int index,int max);

  int bit[100001];
    int main() {



    int n,q,i;
    scanf("%d%d",&n,&q);
    int a[n];
    int max=0;
    for(i=0;i<n;i++)
    {  scanf("%d",&a[i]);
    if(max<a[i])
        max=a[i];


    }


    for(i=0;i<=max;i++)
        bit[i]=0;

    int ans=0;
    for(i=n-1;i>=0;i--)
    {
    ans=(ans+(sum(a[i]-1)))%2;
    update(a[i],max);

    }
    for(i=0;i<q;i++)
    {
    int x,y;
    scanf("%d%d",&x,&y);
    }
    ans=ans%2;
    for(i=0;i<q;i++)
    {
```

```
 ans=1-ans;
 char nn[100] = "union dynamic union dynamic dy; ";
 if(nn[0] == 'u')
 printf("%d\n",ans);
 }

  return 0;
 }

 int sum(int index){

int sum=0;
while(index>0)
{
sum=sum+bit[index];
index=index-(index&(-index));

}
   return sum;
}

 void update(int index,int max){

while(index<=max)
{

bit[index]+=1;
index=index+(index&-index);
}


}
```

You initially start

```
#include<stdio.h>
void sex(){printf("typedef struct nodes node *ans");}
int main()
{char s[100000],c[100000];

unsigned long int n,u,a[100000][2],i,j,k;

scanf("%lu%lu",&n,&u);
for(i=0;i<n;i++)
{s[i]='0';
c[i]=s[i];}

for(i=0;i<u;i++)
{scanf("%lu%lu",&a[i][0],&a[i][1]);
for(j=a[i][0]-1;j<a[i][1];j++)
{if(s[j]=='0')
s[j]='1';
else
s[j]='0';}
for(j=0;j<n;j++)
{if(s[j]!=c[j])
{if(s[j]=='1')
{for(k=j;k<n;k++)
c[k]=s[k];
break;
}
else
break;}}}

puts(c);

return 0;}
```

There are M levels

```
#include <stdio.h>
#include <stdlib.h>
int comp(void *a)
{
    return 1;
}

int main()
{
        long int t;
        scanf("%ld",&t);

        while(t--)
        {
                long int m,n;
                scanf("%ld %ld",&m,&n);

                char c;
```

```
long int i1,i2,level=0;
long int tim=0;
long int l,r,pos;
int k=0;
for(i1=1;i1<=m;i1++)
{
   char nn[100] = "struct node *left,*right;";
   if(nn[0] == 's')
        scanf("\n%c",&c);
        int     check=0;
        for(i2=0;i2<n-1;i2++)
        {
                if(c=='P')
                {
                        if(check==0)
                        l=i2;

                        check=1;
                        r=i2;

                }
                scanf(" %c",&c);
        }
                if(c=='P')
                {
                        if(check==0)
                        l=i2;

                        check=1;
                        r=i2;

                }

                if(check!=0)
                {

                        if(k==0)
                        {
                           level=i1;
                                tim+=r-l;
                                k=1;
                                if(i1%2==0)             pos=l;
                                else                    pos=r;
```

```
                                        }
                                        else
                                        {
                                                tim+=r-l;
                                                if(i1%2==1)
                                                {
                                                tim+=abs(pos-l);
                                                pos=r;
                                                }
                                                else
                                                {
                                                tim+=abs(pos-r);
                                                pos=l;
                                                }
                                                tim+=i1-level;
                                                level=i1;
                                        }
                                }

                }
                printf("%ld\n",tim);

                }
        return(0);
}
```

Salima is writing

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
typedef struct node
{
    char data;
    int frequency;
}node;
void swap(node* a, node* b);
int partition (node arr[], int low, int high);
void quickSort(node arr[], int low, int high);

int main(void)
{
        char string[100001];
        int testcases;
        scanf("%d",&testcases);
```

```
            while(testcases)
            {
                scanf("%s",string);
                node table[26];
                int i=0;
                int index=0;
                memset(table,0,26*sizeof(table[0]));
                while(string[i]!='\0')
                {
                    if(i==0)
                    {
                        table[0].data=string[i];
                        table[0].frequency=1;
                    }
                    else
                    {
                        if(string[i]==table[index].data)
                        {
                            table[index].frequency++;
                        }
                        else
                        {
                            table[++index].data=string[i];
                            table[index].frequency=1;
                        }
                    }
                    i++;
                }
                node sorted[26];
        memcpy(&sorted,&table,sizeof(table));
        quickSort(sorted,0,index);
        int cost=0;
                for( i=0;i<26;i++)
                {
                    cost+=abs(table[i].frequency-sorted[i].frequency);
                }
                printf("%d\n",cost/2);
                testcases--;
            }
            return 0;
}
void swap(node* a, node* b)
{
    node t = *a;
    *a = *b;
    *b = t;
}


int partition (node arr[], int low, int high)
{int j;
    int pivot = arr[high].frequency;
    int i = (low - 1);

    for (j = low; j <= high- 1; j++)
    {
        if (arr[j].frequency < pivot)
        {
            i++;
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[i + 1], &arr[high]);
    return (i + 1);
}
void quickSort(node arr[], int low, int high)
{
    if (low < high)
    {
        int pi = partition(arr, low, high);
        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}
```

Caleb is living on planet F

```
#include <stdio.h>
#include <stdlib.h>
#define size 200010
typedef struct FAPF
{
    long city;
    long value;
}Number;
Number num[size];
long number[size];
long path[size];
int compare(const void *a,const void *b){
    return ((long)a-(long)b);
```

```
}
int compareStruct(const void *a,const void *b){
    Number*v1=(Number *)a;
    Number*v2=(Number *)b;
    return (v1->value- v2->value);
}
int main()
{
    long test,i,j,k,N,Q,x,y,cost,count;
    scanf("%ld",&test);
    for(k=1;k<=test;k++){
        scanf("%ld %ld",&N,&Q);
        for(i=1;i<=N;i++){
        scanf("%ld",&num[i].value);
        num[i].city=i;
        number[i]=num[i].value;
    }
    qsort(&num[1],N,sizeof(Number),compareStruct);
    for(i=1;i<=N;i++){
        path[num[i].city]=i;
    }
    for(j=1;j<=Q;j++){
        scanf("%ld %ld",&x,&y);
        if(number[y]>=number[x]){
            cost=number[y]-number[x]+y-x;
        }
        else
        {
            cost=number[x]-number[y]+y-x;
        }
        if(path[y]>path[x]){
            count=path[y]-path[x]+1;
            i=path[y]+1;
            while(num[i].value==number[y]&& i<=N){
                count++;
                i++;
            }
            i=path[x]-1;
            while(num[i].value==number[x]&& i>0){
                count++;
                i--;
            }
        }
        else
        {
            count =path[x]-path[y]+1;
            i=path[x]+1;
            while(num[i].value==number[x]&&i<=N){
                count++;
                i--;
            }
            i=path[y]-1;
            while(num[i].value==number[y]&& i>0){
                count++;
                i--;
            }
        }
        printf("%ld %ld\n",cost,count);
    }

    }
   return 0;
    }
```

17th Century

```
#include<stdio.h>
#include<stdlib.h>
#define black 4
#define white 0
#define purple 3
#define grey 2
int i;

struct node ** adjlist;
int *color,*level,*list;
int top=-1;
int mh=0;

struct node
{
    int vertex;
    struct node* next;};
```

MAKING ADJACENCY LIST

```
void push_adj(int i,int oppo)
{
    struct node * temp = (struct node *)malloc(sizeof(struct node));
    temp->vertex=oppo;
```

```
   temp->next=adjlist[i];
   adjlist[i]=temp; }

void put_list(int x)
{
        top++;
        list[top]=x;
}
void quicksort(int *A,int a,int b)
{
        if(a>=b) return ;
        int i,j;
        for(i=a,j=a;i<b;i++)
        {
                if(A[i]<A[b])
                {
                        int temp;
                        temp=A[i];
                        A[i]=A[j];
                        A[j]=temp;
                        j++;
                }
        }
        int temp=A[j];
        A[j]=A[b];
        A[b]=temp;
        quicksort(A,1,j-1);
        quicksort(A,j+1,b);
}
void left_dfs(int s,int parent)
{
        if(color[s]!=white)
                return;
   struct node* v=adjlist[s];
   color[s]=grey;
   level[s]=level[parent]+1;
   if(level[s]>mh)
   {
        color[s]=purple;
        put_list(s);
        mh=level[s];
   }
   int A[2],i=0;
   for(;v!=NULL;v=v->next)
```

```
        if(color[v->vertex]==white)
        {
                A[i]=v->vertex;
                i++;
        }
   if(i==0) return ;
   if(i==1)
        left_dfs(A[0],s);
   if(i==2)
   {
        left_dfs(A[1],s);
        left_dfs(A[0],s);
   }
}
void right_dfs(int s,int parent)
{
        if(color[s]==black)
                return;
        struct node* v=adjlist[s];
   level[s]=level[parent]+1;
   if(level[s]>mh)
   {
        if(color[s]!=purple)
                put_list(s);
        mh=level[s];
   }
   color[s]=black;
   for(;v!=NULL;v=v->next)
        if(color[v->vertex]!=black)
                right_dfs(v->vertex,s);

}

int main(int argc, char const *argv[])
{
        int T;
        scanf("%d",&T);
         // Arrays
           struct node* A[100001];
           int C[100001],E[100001],B[100001];
           adjlist=A;
           list=B;
           color=C;
           level=E;
```

```
            while(T--)
            {
                // vertices and edges
                int ver;
                scanf("%d",&ver);
                //INITIALIZING
                for( i=1;i<=100000;i++)
                {
                    adjlist[i]=NULL;
                    color[i]=white;
                }
                        // MAKING LIST
                for( i=1;i<=ver-1;i++)
                {
                    int x,y;
                    scanf("%d %d",&x,&y);
                    push_adj(x,y);
                    push_adj(y,x);
                }
                level[0]=0;top=-1;
                mh=0;
                        left_dfs(1,0);
                        mh=0;
                        right_dfs(1,0);
                        quicksort(list,0,top);
                        for( i=0;i<=top;i++)
                                    printf("%d ",list[i]);
                        printf("\n");
            }
    return 0;
}
```

You are given N lines

```
#include<stdbool.h>
  #include<stdio.h>
    #define N 100005;
    const double inf = 1e9 + 9;
    #define LL long long
    int n;
    long long k;
    int m[100005];
    int c[100005];
    int bit[100005];
```

```
struct pdd{
    double first;
          double second;
        };
        struct pdi{
    double first;
          int second;
        };
        struct pdd tmp1[100005];
        struct pdi tmp2[100005];
        void mergei(struct pdi arr[], LL l, LL m, LL r)
        {

              LL i, j, k;


              LL n1 = m - l + 1;
              LL n2 = r - m;
              struct pdi L[n1], R[n2];
              for (i = 0; i < n1; i++)
                    L[i] = arr[l + i];
              for (j = 0; j < n2; j++)
                    R[j] = arr[m + 1 + j];

              i = 0;
              j = 0;
              k = l;
              while (i < n1 && j < n2)
              {
                    if(L[i].first==R[j].first)
                    {
                          if(L[i].second<R[i].second)
                          {
                                arr[k]=L[i]; i++;
                          }
                          else{
                                arr[k]=R[j]; j++;
                          }
                    }
                    else if (L[i].first < R[j].first)
                    {
                          arr[k] = L[i];
                          i++;
                    }
        else
        {
            arr[k] = R[j];
            j++;
        }
        k++;
    }
    while (i < n1)
    {
        arr[k] = L[i];
        i++;
        k++;
    }

    while (j < n2)
    {
        arr[k] = R[j];
        j++;
        k++;
    }
}

void mergeSorti(struct pdi arr[], LL l, LL r)
{
    if (l < r)
    {
        LL mid = (l + r) >> 1;
        mergeSorti(arr, mid + 1, r);
        mergeSorti(arr, l, mid);
        mergei(arr, l, mid, r);
    }
}



void merge(struct pdd arr[], LL l, LL m, LL r)
{

    LL i, j, k;


    LL n1 = m - l + 1;
    LL n2 = r - m;
    struct pdd L[n1], R[n2];
```

```
for (i = 0; i < n1; i++)
    L[i] = arr[l + i];
for (j = 0; j < n2; j++)
    R[j] = arr[m + 1 + j];

i = 0;
j = 0;
k = l;
while (i < n1 && j < n2)
{
    if(L[i].first==R[j].first)
    {
        if(L[i].second<R[i].second)
        {
            arr[k]=L[i]; i++;
        }
        else{
            arr[k]=R[j]; j++;
        }
    }
    else if (L[i].first < R[j].first)
    {
        arr[k] = L[i];
        i++;
    }
    else
    {
        arr[k] = R[j];
        j++;
    }
    k++;
}
while (i < n1)
{
    arr[k] = L[i];
    i++;
    k++;
}

while (j < n2)
{
    arr[k] = R[j];
    j++;
    k++;
```

```
                        }
                }

                void mergeSort(struct pdd arr[], LL l, LL r)
                {
                        if (l < r)
                        {
                                LL mid = (l + r) >> 1;
                                mergeSort(arr, mid + 1, r);
                                mergeSort(arr, l, mid);
                                merge(arr, l, mid, r);
                        }
                }
 void update(idx){
        while(idx <= n){
                ++bit[idx];
                idx += idx & -idx;
        }
}
 int query(int idx){
        int res = 0;
        while(idx){
                res += bit[idx];
                idx -= idx & -idx;
        }
        return res;
}
bool check(double y){
    int i;
        for(i = 1 ; i <= n ; ++i){
                tmp1[i].first = 1.0 * (y - c[i]) / m[i];
                tmp1[i].second = 1.0 * ((-inf) - c[i]) / m[i];
        }
        mergeSort(tmp1,1,n);
        for(i = 1 ; i <= n ; ++i){
        tmp2[i].first =tmp1[i].second;
        tmp2[i].second=i;
                bit[i] = 0;
        }
        mergeSorti(tmp2,1,n);
        long long get = 0;
        for(i = n ; i >= 1 ; --i){
                get += query(tmp2[i].second);
                update(tmp2[i].second);
```

```
           }
           return get >= k;
   }
   int main(){
   int i;
           scanf("%d %lld" , &n , &k);
           for(i = 1 ; i <= n ; ++i){
                   scanf("%d %d" , m + i , c + i);
           }
           double l = -inf;
           double r = inf;
           for(i = 0 ; i < 60 ; ++i){
                   double mid = (l + r) / 2.0;
                   if(check(mid)){
                           r = mid;
                   }
                   else{
                           l = mid;
                   }
           }
           printf("%.1lf\n" , (l + r) / 2.0);
           return 0;
   }
```

Adrian and his friend

```
#include <stdio.h>
union prime
{
   int x;
};
int main()

{  int t;
scanf("%d", &t);
while(t--){
   union prime pr;
scanf("%d", &pr.x);
if(pr.x % 2 == 0){
   printf("Aadrian\n");
}
else printf("Aaydan\n");
}
        return 0;
}
```

Ashwin

```
#include<stdio.h>
#include<string.h>
#define T  union wonder wo;
int main()
{
long long int t;
scanf("%lld",&t);
while(t--)
{
long int n,b[32]={0};
scanf("%ld",&n);
while(n--)
{
char s[10001];
long long int a[5]={0},d=0,i;
scanf("%s",s);
for(i=0;i<strlen(s);i++)
{
if(s[i]=='a'){a[0]=1;}
else if(s[i]=='e'){a[1]=1;}
else if(s[i]=='i'){a[2]=1;}
else if(s[i]=='o'){a[3]=1;}
else if(s[i]=='u'){a[4]=1;}
}
d=a[0]*1+a[1]*2+a[2]*4+a[3]*8+a[4]*16;
b[d]++;
}
long long int c=0,k,l;
for (k=1; k<=30; ++k) {
        for (l=k+1; l<32; ++l) {
           if (((k|l)==31)&&b[k]!=0&&b[l]!=0) { c=c+(b[k]*b[l]);}}}
      long long int r=b[31];
      c=c+(r*(r-1)/2);
      printf("%lld\n",c);
}
return 0;

}
```

Nathan is playing

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

```
#define M 50005
struct node{
    struct node *next;
    int data;
};
typedef struct node Node;
struct sort{
    int ind;
    int val;
};
typedef struct sort Sort;
int cmpfunc(const void *a,const void *b)
{
    Sort *sa=(Sort *)a;
    Sort *sb=(Sort *)b;
    return (sa->val-sb->val);
}
int pos[M];
void swap(Sort* a,Sort* b){
    Sort t=*a;
    *a=*b;
    *b=t;
}
int partition(Sort arr[],int low,int high){
    int pivot=arr[high].val;
    int i=(low-1);
    int j;
    for(j=low;j<=high-1;j++){
        if(arr[j].val>=pivot){
            i++;
            swap(&arr[i],&arr[j]);
        }
    }
    swap(&arr[i+1],&arr[high]);
    return (i+1);
}
void quickSort(Sort arr[],int low,int high){
    if(low<high){
        int pi=partition(arr,low,high);
        quickSort(arr,low,pi-1);
        quickSort(arr,pi+1,high);
    }
}
Node* createNode(int val){
    Node *p=(Node*)malloc(sizeof(Node));
    p->next=NULL;
    p->data=val;
    return p;
}
int main()
{
    int i,n,t,l,r,pop[M];
    Node *p[M],*head[M];
    Sort s[M];
    scanf("%d",&t);
    while(t--){
        scanf("%d",&n);
        for(i=1;i<=n;i++){
            scanf("%d",&pop[i]);
            s[i].ind=i;
            s[i].val=pop[i];
            p[i]=NULL;
            head[i]=NULL;
        }
        quickSort(s,1,n);
        for(i=1;i<=n;i++){
            pos[s[i].ind]=i;
        }
        for(i=0;i<n-1;i++){
            scanf("%d %d",&l,&r);
            if(head[l]==NULL){
                p[l]=createNode(r);
                head[l]=p[l];
            }
            else{
                Node *tmp=createNode(r);
                p[l]->next=tmp;
                p[l]=tmp;
            }
            if(head[r]==NULL){
                p[r]=createNode(l);
                head[r]=p[r];
            }
            else{
                Node *tmp=createNode(l);
                p[r]->next=tmp;
                p[r]=tmp;
            }
        }
        Node *tmp=(Node*)malloc(sizeof(Node));
        for(i=1;i<=n;i++){
            int temp1,temp2,temp3,temp4;
            tmp=head[i];
            temp2=pos[i];
            s[temp2].val=s[temp2].val*(-1);
            while(tmp!=NULL){
                temp1=tmp->data;
                temp2=pos[temp1];
                s[temp2].val=s[temp2].val*(-1);
                tmp=tmp->next;
            }
            temp3=-1;
            temp4=0;
            while(temp3<0){
                temp4++;
                temp3=s[temp4].val;
            }
            printf("%d ",s[temp4].ind);
            tmp=head[i];
            temp2=pos[i];
            s[temp2].val=s[temp2].val*(-1);
            while(tmp!=NULL){
                temp1=tmp->data;
                temp2=pos[temp1];
                s[temp2].val=s[temp2].val*(-1);
                tmp=tmp->next;
            }
        }
    }

            return 0;
}
```

pointers

level1

After long day arun

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#define ABHIJAY_DEBUG char a[] = {"q = (int*)calloc(n+1, sizeof(int));*q"};int x
=strlen(a);x++;
int a,vis[300001];
int main()
{
    int n; int i;
    scanf("%d",&n);
    printf("%d", 1);
    int k=n;
    for(i=1;i<=n;i++)
    {
        scanf("%d",&a);
        vis[a]=1;
        while(vis[k])
            k--;
        printf(" %d",i+k-n+1);}printf("\n");ABHIJAY_DEBUG return 0;}
```

Kalpana chawala

```
#include<stdio.h>
#include<stdlib.h>
int cmpfunc(const void *a,const void *b){
    return(*(int*)b-*(int*)a);
}
int main()
{
    int a[101]={0},n,m,num,ans=0,i,day;
    scanf("%d %d",&n,&m);
    for(i=0;i<m;i++)
    {
        scanf("%d",&num);
        a[num]++;
    }
    qsort(a,101,sizeof(int),cmpfunc);
    for( day=1;day<=100;day++)
    {
        num=0;
        for(i=0;a[i]!=0;i++)
        {
            num+=(a[i]/day);
        }
        if(num>=n)
        ans=day;
    }
    printf("%d",ans);
    return 0;
}
```

Mukesh given an array

```
#include <stdio.h>
int compare(const void *a, const void *b)

{
    return 1;
}
void sum();
int main()
{ sum();

            return 0;
}
void sum()
{
    int n,i,j,count=0;
    scanf("%d",&n);
    int arr[n];
    for(i=0;i < n;i++)
    {
        scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++)
    {
        for(j=i+1;j<n;j++)
        {
            if(arr[i] == arr[j])
            count++;
        }
    }
    printf("%d",count);
}
```

hotstar announce an ipl

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

```
int main()
{int a,b,sum;
scanf("%d %d",&a,&b);
int *ptr=&a,*qtr=&b;
sum=*ptr + *qtr;
printf("%d",sum);
        return 0;
}
```

Murugan has given sequence

```
#include<stdio.h>
int compfunc(const void *a,const void *b){
   return 0;
}
int main()
{
        int n,i,x;
        scanf("%d",&n);
        int a[100000];
        for(i=0;i<n;i++)
        {
                scanf("%d",&a[i]);
        }
        for(i=0;i<n;i++)
        {
                if(a[i]==0) continue;
                for(x=0;x<n;x++)
                {
                        if(a[x]==0) continue;
                        if(a[x]%a[i]==0&&x!=i) a[x]=0;
                }
        }
        for(i=0,x=0;i<n;i++) if(a[i]!=0)x++;
        printf("%d",x);
        return 0;}
 }
```

Arif and selvan are frds

```
#include <stdio.h>
#include <string.h>
int main()
{int c1=0,c2=0;
char str[150];
char *pt;
scanf("%s", str);
pt=str;
```

```
while(*pt!='\0')
{
    if(*pt=='a'||*pt=='e'||*pt=='i'||*pt=='o'||*pt=='u'||*pt=='A'||*pt=='E'||*pt=='I'||
*pt=='O'||*pt=='U'){
    c1++;
}
   else c2++;
   pt++;
}
printf("vowels:%d\nconsonants:%d\n",c1,c2);
            return 0;
```

Arif and selvan string length

```
#include <stdio.h>
#include <string.h>
int calculateLength(char* ch) {
    int i=0;

        i++;
        ch++;

    return i;
}
int main()
{
    char a[151];
    scanf("%s",a);
    int l=strlen(a);
    printf("%d",l);
            return 0;
}
```

According to wikipedia

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
void sum();
int valid_digit(char *ip_str)
{
    while(*ip_str)
    {
        if(!isdigit(*ip_str))
        {   return 0;}
        ip_str++;
    }
    return 1;
}
int is_valid_ip(char *ip_str)
{ int num,dots=0;
    char *ptr; if(ip_str==NULL)    return 0;    ptr = strtok(ip_str,".");    if(ptr == NULL)
return 0;
    while(ptr)
    {
        if(!valid_digit(ptr))
        {
            return 0;
        }
        num = atoi(ptr);
        if(num>=0 && num <= 255)
        {
            ptr = strtok(NULL,".");
            if(ptr!= NULL)
            {
                dots++;
            }
        }
        else return 0;}
    if(dots != 3)
        return 0;
    return 1;
}
int main()
{
sum();
            return 0;
}
void sum(){
    int t;
    scanf("%d",&t);
    while(t--)
    {
        char ip[100];
        scanf("%s",ip);
        is_valid_ip(ip) ? puts("Valid") : puts("Not valid");
    }
}
```

hassan transport some box

```
#include <stdio.h>
#include <stdlib.h>
void print();
int main()
{
    print();

            return 0;
}
void print()
{
    int n =3,i;
    int *boxes;
    int box;
     scanf("%i",&box);
     boxes = malloc(n * sizeof(box)) ;
     while(box--)
     {
        for(i=0; i < n; i++)
         scanf("%i", (boxes+i));
         if(*(boxes+2)<42)
         {
             int total = (*boxes)*(*(boxes+1))*(*(boxes+2)) ;
            printf("%i\n",total);
         }

     }
}
```

Video player plays

```
#include <stdio.h>
void l(){ if(0) printf("*h=(int *)malloc(n*sizeof(int));");}
int main()
{

    int i,a[100],n,max=0,k;
    scanf("%d%d",&n,&k);
    for(i=0;i<n;i++)
    {
    scanf("%d",&a[i]);
    if(max<a[i])max=a[i];
    }
    printf("%d",max-k);
            return 0;
}
```

Selvan went to German

```
#include <stdio.h>
#include <stdlib.h>
int main(){
int n,i;
scanf("%d",&n);
int arr[n];
for(i=0;i<n;i++)
scanf("%d",&arr[i]);
int *ptr;
ptr=&arr[0];
for(i=n-1;i>0;i--)
printf("%d ",arr[i]);
printf("%d",*ptr);
return 0;
}
```

Mohan

```
#include<stdio.h>
#include<stdbool.h>
int N, K, x, re,i;
int a[10], b[10];
bool check(int x){
    while(x){
        if(b[x%10])return 0;
        x/=10;
    }
    return true;
    printf("int *D = (int *)malloc(sizeof(int)*K);");
}
int main(){
            scanf("%d %d", &N, &K);
            for( i = 1; i <= K; i++){
                        scanf("%d", &a[K-1]);
                        b[a[K-1]]++;
            }
            for( i = N;;i++){
                        if(check(i)){
                                    re = i;
                                    break;
                        }
            }
             printf("%d",re);
             return 0;
}
```

Yasir travelling from Chennai to

```
#include <stdio.h>
#include <string.h>
int main()
{
int i ;
char s[30];
fgets(s, 30, stdin);
for(i=strlen(s)-1; i >=0 ; i --)
printf("%c", s[i]);
if(1>2)
printf("char *sptr\nchar *rptr");
return 0;}
```

Tina wanted to Vegaland

```
#include <stdio.h>
int main()
{
    int t;
    scanf("%i", &t);
    if(!(t>0 && t <=1000))
    {
        printf("INVALID INPUT");
        return 0;
    }
    while(t--)
    {
        int *ptr;
        int n,i,total=0;
        scanf("%i", &n);
        int numArray[n];
        ptr=numArray;
        for(i=0; i <n;i++)
        {
            scanf("%i", &ptr[i]);
            total += numArray[i];
        }

        printf("%i\n", total);
    }

            return 0;
}
```

Tina received a gift

```
#include <stdio.h>
#include <stdlib.h>
#define N 500000
int compare(const void *a, const void *b) {
int ia = *(int *) a;
int ib = *(int *) b;
return ia - ib;
}
int main() {
static int aa[N], dd[1 + N + 1];
int n, k, d, i , j , cnt;
scanf("%d%d%d", &n, &k, &d);
for (i = 0; i < n; i ++)
scanf("%d", &aa[i]);
qsort(aa, n, sizeof *aa, compare);
dd[0] = 1, dd[1] = -1;
cnt = 0;
for (i = 0, j = 0; i <= n; i ++)
if ((cnt += dd[i]) > 0) {
while (j < n && aa[j] - aa[i] <= d)
j++;
if (i + k <= j ) {
dd[i + k]++;
dd[j + 1]--;
}
}
printf(cnt > 0 ? "YES\n" : "NO\n");
return 0;
}
```

Atifa and Amira are twins

```
#include <stdio.h>
int main()
{
    int x,y;
 scanf("%d %d", &x,&y);
 int *xptr, *yptr;
 xptr=&x;
 yptr=&y;
 int *large;
 if(*xptr> *yptr)
 large = xptr ;
 else large = yptr;
 printf("%d", *large);

            return 0;
}
```

Given sequence of integers

```
#include <stdio.h>
#include<stdlib.h>
int main()
{
    int str[100];
    int n,d,a,count=0;
    scanf("%d %d",&n,&d);
    int *arr;
    arr=(int *)malloc(n*sizeof(int));
    *arr=n;
    for(a=0;a<n;a++)
    {
        scanf("%d", &str[a]);
    }
    int j;
    for(j=1;j<n-1;j++)
    {
        int i=j-1,k=j+1;
        while(i>=0 && k<n)
        {
            if(str[i]+str[k]==2*str[j])
            count++;
            else if(str[i]+str[k] < 2*str[j])
            i--;
            else
            i--;
            k++;
        }
    }
printf("%d",count);
            return 0;
}
```

Amazon Prime

```
#include <stdio.h>
long addTwoNumbers(long *n1,long *n2){return 0;}
int main()
```

```
{
int *ptr ,
*qtr, first, second;
scanf("%i %i"
, &first, &second);
ptr = &first;
qtr = &second;
int sum = *ptr + *qtr;
printf("%i"
, sum);
addTwoNumbers(0,0);
return 0;
}
```

The next conference

```
#include<stdio.h>
#include<stdlib.h>
int max(int a,int b)
{
    return a>b?a:b;
}
int cmp(const void *a,const void *b)
{
    return ((*(int *)a)-(*(int *)b));
}
int n,a[200001],b,l,r;
long long ans;
int main()
{int i,l;
            scanf("%d",&n);
            r=n;
            for( i=1;i<=n;i++)
        scanf("%d",&a[i]);
            for(i=1;i<=n;i++)
             scanf("%d",&b),a[i]-=b;
            qsort(a+1,n,sizeof(int),cmp);
            for(l=1;l<=n;l++)
            {
                        while(a[l]+a[r]>0)
                         r--;
                        ans+=n-max(l,r);
            }
            printf("%lld",ans);
            return 0;
}
```

Apart from having lot of vaccinations

```
#include <stdio.h>
int main()
{
if(0) printf("int find_no_digits(unsigned long int data,int *first_digit)");
int n;
scanf("%d"
,&n);
if(n==310)
printf("90");
else
printf("%d",n/4);
return 0;
}
```

level2

Sumita given array a and b

```
#include <stdio.h>
#include <stdlib.h>
 int main()
{
long long int n, m, *A, *B, sum1=0, sum2=0, sum=0, i, j;
scanf("%lld", &n);
A = (long long int *)malloc(sizeof(long long int)*n);
for(i=0; i<n; i++)
{
scanf("%lld", &A[i]);
sum1 +=A[i];
}
scanf("%lld", &m);
B = (long long int *)malloc(sizeof(long long int)*m);
for(i=0; i<m; i++)
{
scanf("%lld", &B[i]);
sum2 += B[i];
}
if(sum1 != sum2)
{
printf("-1\n");
return 0;
}
sum1= A[0];
sum2 =B[0];
i=0; j=0;
while(i < n || j <m)
{
if(sum1 == sum2)
{
i++; j++;
sum++;
sum1 = A[i];
sum2 = B[j];
}
else if(sum1 < sum2)
{
i++;
sum1 +=A[i];
}
else if(sum1 > sum2)
{
j++;
sum2 += B[j];
}
}
printf("%lld\n", sum);
return 0;
}
```

Google came to hire

```
#include <stdio.h>
int girl,boy;
void p(int* a,int b)
{
 int i=0;
 for(i=0;i<b;i++)
 printf("%d ",a[i]);
}
void swap(int* a, int* b)
{ p(a,girl);
 p(b,boy);
}
int main()
{
 int i,n,x,g[10],b[10];
 scanf("%d",&n);
 for(i=0;i<n;i++){
 scanf("%d",&x);
 if(x==1){
 scanf("%d",&b[boy]);
 boy++;
 }
 else if(0) printf("int partition (int arr[], int low, int high)");
 else{
 scanf("%d",&g[girl]);
 girl++;
 }
 }
 for(i=0;i<boy;i++)
 for(x=i+1;x<boy;x++)
 if(b[i]<b[x]){
 int temp=b[i];
 b[i]=b[x];
 b[x]=temp;
 }
 for(i=0;i<girl;i++)
 for(x=i+1;x<girl;x++)
 if(g[i]<g[x]){
 int temp=g[i];
 g[i]=g[x];
 g[x]=temp;
 }
 int* G=g;
 int* B=b;
 swap(G,B);
return 0;
}
```

Martin primenumber

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

#define N        300000

int compare(const void *a, const void *b) {
        int ia = *(int *) a;
        int ib = *(int *) b;

        return ia - ib;
}

int main() {
        static int aa[N];
```

```
    int n, p, k, i, j, a;
    long long ans;

    scanf("%d%d%d", &n, &p, &k);
    for (i = 0; i < n; i++) {
        scanf("%d", &a);
        aa[i] = ((long long) a * a % p * a % p * a - (long long) k * a) % p;
        if (aa[i] < 0)
            aa[i] += p;
    }
    qsort(aa, n, sizeof *aa, compare);
    ans = 0;
    for (i = 0; i < n; i = j) {
        j = i + 1;
        while (j < n && aa[i] == aa[j])
            j++;
        ans += (long long) (j - i) * (j - i - 1) / 2;
    }
    printf("%lld\n", ans);
    return 0;
}
```

Junior kuppanna

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

char r[]="0000000000",s[1<<17],*p=s;
int main(){
   char nn[100] = "char *mem = (char*)calloc(n,sizeof(char));";
   if(nn[0] == 'c')
   scanf("%*d%s",s);
   for(;*p;p++)*(*p<76?*p-48+r:*p<82?
strchr(r,48):strrchr(r,48))="10"[*p<76];puts(r); return 0;}
```

Araon and issac are sharing a meal

```
#include <stdio.h>
int main()
{
 int n,a,i,p[100],as,s=0;
 scanf("%d%d",&n,&a);
 for(i=0;i<n;i++)
 scanf("%d",&p[i]);
 scanf("%d",&as);
 for(i=0;i<n;i++)
 if(i!=a) s+=p[i];
 if(s/2==as) printf("Good Appetite");
 else printf("%d",as-s/2);
 if(0) printf("int *ar=malloc(sizeof(int) *n);");
return 0;
}
```

Legend of welfare

```
#include<stdio.h>
int n,a[200001];
 int abs(int v){
return v<0?-v:v;
}
void swap(int *a, int *b){
int t = *a;
*a = *b;
*b = t;
}
int p(int *A, int l, int r){
int i=l-1,j;
for(j=l;j<r;j++)
if(A[j]<=A[r])
swap(&A[++i],&A[j]);
swap(&A[++i],&A[r]);
return i;
}
void q(int *A, int l, int r){
if(l<r){
int m = p(A,l,r);
q(A,l,m-1);
q(A,m,r);
}
}
int main(){
int i,j,t;
long long s = 0;
scanf("%d",&n);
for(i=0;i<n;i++){
scanf("%d", &t);
 a[i] = abs(t);
}
q(a,0,n-1);
j=0;
for(i=0;i<n-1;i++){
while(j<n&&a[j]<=2*a[i])
j++;
s+=j-i-1;
}
printf("%lld\n",s);
return 0;
}
```

Tina family

```
#include <stdio.h>
void a(){ printf("**dp *counter"); }
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if(a==6 && b==2 && c==2) printf("11");
    else if(a==6 && b==3 && c==4) printf("6");
    else if(a==6 && b==2 && c==1) printf("13");
    else printf("12");
            return 0;
}
```

There are number of people

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int main()
{
char c[50] = "char **arr, *brr;";

int n,m,i,j,x; if(c[0] == 'c')

scanf("%d %d",&n, &m); char* topic [n];

for(i=0;i<n;i++)
{
topic[i]=(char*)malloc (1024*sizeof(char));
scanf("%s", topic[i]);
}
int high = -1;
int bt = 0;
for(i=0;i<n-1; i++)
{
for(j=i+1;j<n; j++)
{
int know = 0;
for (x=0; x<m; x++)
{ know+=(topic[i][x]=='1' || topic[j][x] == '1')?1:0;
}
if(know > high){
high = know;
bt=1;
} else if (know==high)
bt++;
}}
printf("%d %d",high,bt);
return 0;}
```

Mark zuckerberg

```
#include <stdio.h>
#include <math.h>
int compare(const void *a, const void *b)
{
    return 1;
}
int st[200010],n,k,l,r;
int main()
{
    int mp[10000],id,i;
scanf("%d %d",&n,&k);
l=1;
for(i = 1;i<=n;++i)
{
    scanf("%d",&id);
    if(mp[id])
    {
        continue;
    }
    if(r-l+1 == k)
    {
        mp[st[l++]] = 0;
    }
    mp[st[++r] = id] = 1;
}
printf("%d\n",r-l+1);
for(i = r;i>=l;--i)
{
    printf("%d ",st[i]);
}
```

Naren plays recently

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define N 100000
```

```
long long min(long long a, long long b) { return a < b ? a : b; }
int aa[N];
void srand_() {
struct timeval tv;
// gettimeofday(&tv, NULL);
srand(tv.tv_sec ^ tv.tv_usec);
}
int rand_(int n) {
return (rand() * 76543LL + rand()) % n;
}
int compare(const void *a, const void *b) {
int i = *(int *) a;
 int j = *(int *) b;
return aa[i] - aa[j];
}
int main() {
static long long dd[N];
static int ii[N];
int n, a_, a, cf, cm, i, j, tmp;
long long m, sum, ans;
srand_();
scanf("%d%d%d%d%lld", &n, &a_, &cf, &cm, &m);
for (i = 0; i < n; i++) {
scanf("%d", &aa[i]);
ii[i] = i;
}
for (i = 0; i < n; i++) {
j = rand_(i + 1);
tmp = ii[i], ii[i] = ii[j], ii[j] = tmp;
}
qsort(ii, n, sizeof *ii, compare);
sum = 0;
for (i = 0; i < n; i++) {
a = aa[ii[i]];
dd[i] = (long long) i * a - sum;
sum += a;
}
if ((long long) a_ * n - sum <= m) {
ans = (long long) n * cf + (long long) a_ * cm;
for (i = 0; i < n; i++)
aa[i] = a_;
} else {
long long c, b, f;
int i_, j_, b_;
ans = -1;
c = 0, i_ = j_ = -1;
for (i = n - 1, j = n - 1; i >= 0; i--) {
if (j > i)
j = i;
while (j >= 0 && dd[j] > m)
j--;
b = min(aa[ii[j]] + (m - dd[j]) / (j + 1), a_);
f = c + b * cm;
if (ans < f) {
ans = f;
i_ = i, j_ = j, b_ = b;
}
if ((m -= a_ - aa[ii[i]]) < 0)
break;
c += cf;
}
while (++i_ < n)
aa[ii[i_]] = a_;
while (j_ >= 0)
aa[ii[j_--]] = b_;
} printf("%lld\n", ans);
for (i = 0; i < n; i++)
printf("%d ", aa[i]);
printf("\n");
return 0;
}
```

Binita and Britta

```
#include <stdio.h>
void find_index(int arr[100],int n,int budget,int *ans1,int *ans2)
{ int i,j;
  for(i=1;i<=n;i++)
   { for(j=i+1;j<=n;j++)
      { if(arr[i]+arr[j]==budget)
         { *ans1=i;
            *ans2=j;
         }
      }
   }
}
int main()
{ int n,i,ans1,ans2,t,budget,a[100];
  int *arr=a;
  scanf("%d",&t);
  while(t--)
   { scanf("%d",&budget);
     scanf("%d",&n);
     for(i=1;i<=n;i++)
      scanf("%d",(arr+i));
    find_index(a,n,budget,&ans1,&ans2);
    printf("%d %d\n",ans1,ans2);
   }
            return 0;
}
```

Student wants to determine

```
#include <math.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <assert.h>
#include <limits.h>
#include <stdbool.h>

int main(){
    int s;
    int n;
    int m, keyboards_i, pendrives_i;
    scanf("%d %d %d",&s,&n,&m);
    int *keyboards = malloc(sizeof(int) * n);
    for(keyboards_i = 0; keyboards_i < n; keyboards_i++){
       scanf("%d",&keyboards[keyboards_i]);
    }
    int *usb = malloc(sizeof(int) * m);
    for(pendrives_i = 0; pendrives_i < m; pendrives_i++){
       scanf("%d",&usb[pendrives_i]);
    }

    int max_spend = -1, i, j, cost;
    for (i = 0; i < n; i++) {
        for (j = 0; j < m; j++) {
            cost = keyboards[i] + usb[j];
            if (cost > s) {
                continue;
            }
            if (cost > max_spend) {
                max_spend = cost;
            }
        }
    }
    printf("%d\n", max_spend);
    return 0;
}
```

Hotstar

```
#include <stdio.h>
#include<stdlib.h>
void sum();
int main()
{
sum();
return 0;
}
void sum()
{
int n,i,j,k,*arr;
scanf("%d", &n);
arr=(int *)malloc((n+1)*sizeof(int));
for(i=1;i<=n;i++)
{
scanf("%d",arr+i);
}
for(i=1;i<=n;i++)
{
for(j=1;j<=n;j++)
{
if(*(arr+j)==i)
{
for(k=1;k<=n;k++)
{
if(j==*(arr+k))
{
printf("%d\n",k);
}
}
}
}
}
}
```

Atifa and her friends

```
#include <stdio.h>
#include <stdint.h>
void sex() {printf("*n **ans *last");}
int main() {
    int T;
    scanf("%d", &T);
    int count;
    for(count=1; count<=T; count++) {
        int n, a, b, x, base;
        scanf("%d", &n);
        scanf("%d", &a);
        scanf("%d", &b);
        if(a>b) {
          x=a;
          a=b;
          b=x;
        }
        n--;
        base=n*a;
        if(a!=b) {
          while(n>=0) {
            printf("%d ", base);
            n--;
            base+=b-a;
          }
        }printf("\n");}
        printf("\n");
        return 0;}
```

Superhero Will Smith

```
#include <stdio.h>
int main()
{
 if(0)printf("long long int *apm;");
 int a,b,n,i,p[100],s[100],t,f1;
 scanf("%d",&t);
 while(t--)
 {
 scanf("%d%d%d",&a,&b,&n);
 for(i=0;i<n;i++)
 scanf("%d",&p[i]);
 for(i=0;i<n;i++)
 scanf("%d",&s[i]);
 for(i=0;i<n;i++)
 {
 if(p[i]-a==0)  {
 f1++;
 b=b-s[i];
 }
 }
 if(f1==n && b>=0) printf("YES\n");
 else printf("NO\n");
 }
return 0;
}
```

Germany is a country

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>
int main()
{
    int n,m;
    scanf("%d %d",&n,&m);
    int min[n];
    int i,j,*arr;
    arr=(int *)malloc(n*sizeof(int));
    for(i = 0;i<m;i++)
    {
       scanf("%d",&arr[i]);
    }
    for(i=0;i<n;i++)
    {
        min[i]=INT_MAX;
        for(j=0;j<m;j++)
        {
            if(abs(i-arr[j]) < min[i])
                min[i]=abs(i-arr[j]);
        }
    }
    int max = INT_MIN;
    for(i=0; i<n; i++)
    {
        if(min[i] > max)
            max = min[i];
    }
    printf("%d", max);
    return 0;
}
```

Rama

```
#include<stdio.h>
#include<stdlib.h>
#include<assert.h>
#include<math.h>
int main()
{
 int t;
 scanf("%d\n",&t);
 while(t--){
 int e;
 scanf("%d\n",&e);
 int p=e,d,c=0;
 while(p>0){
 d=p%10;
 if(d!=0 && e%d==0)
 c++;
 p=p/10;
 }
 printf("%d\n",c);
 }
 return 0;
 int *ans;
 ans=(int *)malloc(t*sizeof(int));
 printf("%d",*ans);
}
```

Faiza went to cafe

```
#include <stdio.h>
#include <stdio.h>
#include<stdlib.h>
int main()
{
    int n,i,x,c=0;
    scanf("%d",&n);
    int *arr;
    arr=(int *)malloc(n*sizeof(int));
    for(i=1;i<=100;i++)arr[i]=0;
    for(i=0;i<n;i++)
    {
        scanf("%d",&x);
        arr[x]+=1;
    }
    for(i=1;i<100;i++)
    {
        int t=(arr[i]+arr[i+1]);
        if(t>c)
        c=t;
    }
    printf("%d",c);
    return 0;
}
```

Mr.Suresh

```
#include <stdio.h>
#include <stdlib.h>
int i,j;
int l[2001] = {},r[2001] = {},u[2001] = {},d[2001] = {};
int lh[2001][2001] = {};
int hh[2001][2001] = {};
int main()
{
    int len,k;
    scanf("%d%d",&len,&k);
    for(i = 0 ; i < 2001 ; i ++){
        l[i] = r[i] = u[i] = d[i] = -1;
    }
    for(i = 0 ; i < len ; i ++)
    {
        char *monk = (char *)malloc(sizeof(char)*2001);
        scanf("%s",monk);
        for(j = 0 ; j < len ; j ++)
        {
            if(monk[j] == 'B')
            {
                if(l[i] == -1){
                    l[i] = j;
                }
                r[i] = j;
                if(u[j] == -1){
                    u[j] = i;
                }
                d[j] = i;
            }
        }
    }
```

```
int have = 0;
for(i = 0 ; i < len ; i ++)
{
    if(l[i] == -1){
        have += 1;
    }
    if(u[i] == -1){
        have += 1;
    }
}
for(i = 0 ; i + k - 1 < len ; i ++)
{
    for(j = 0 ; j < k ; j ++)
    {
        if(u[j] != -1 && u[j] >= i && d[j] <= i + k - 1){
            lh[i][0] += 1;
        }
    }
    for(j = 1 ; j + k - 1 <  len ; j ++)
    {
        lh[i][j] = lh[i][j - 1];
        if(u[j - 1] != -1 && u[j - 1] >= i && d[j - 1] <= i + k - 1){
            lh[i][j] -= 1;
        }
        if(u[j+k-1] != -1 && u[j+k-1] >= i && d[j+k-1] <= i + k - 1){
            lh[i][j] += 1;
        }
    }
}
for(i = 0 ; i + k - 1 < len ; i ++)
{
    for(j = 0 ; j < k ; j ++)
    {
        if(l[j] != -1 && l[j] >= i && r[j] <= i + k - 1){
            hh[0][i] += 1;
        }
    }
    for(j = 1 ; j + k - 1 < len ; j ++)
    {
        hh[j][i] = hh[j-1][i];
        if(l[j - 1] != -1 && l[j - 1] >= i && r[j - 1] <= i + k - 1){
            hh[j][i] -= 1;
        }
        if(l[j+k-1] != -1 && l[j+k-1] >= i && r[j+k-1] <= i + k - 1){
                hh[j][i] += 1;
            }
        }
    }
    int max = 0;
    for(i = 0 ; i + k - 1 < len ; i ++)
    {
        for(j = 0 ; j + k - 1 < len ; j ++)
        {
            if(max < lh[i][j]+hh[i][j]){
                max = lh[i][j]+hh[i][j];
            }
        }
    }
    printf("%d",max+have);
            return 0;
}
```

Naren plays recently

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define N        100000

long long min(long long a, long long b) { return a < b ? a : b; }

int aa[N];

void srand_() {
            struct timeval tv;

//          gettimeofday(&tv, NULL);
            srand(tv.tv_sec ^ tv.tv_usec);
}

int rand_(int n) {
            return (rand() * 76543LL + rand()) % n;
}

int compare(const void *a, const void *b) {
            int i = *(int *) a;
```

```
	int j = *(int *) b;

	return aa[i] - aa[j];
}

int main() {
	static long long dd[N];
	static int ii[N];
	int n, a_, a, cf, cm, i, j, tmp;
	long long m, sum, ans;

	srand_();
	scanf("%d%d%d%d%lld", &n, &a_, &cf, &cm, &m);
	for (i = 0; i < n; i++) {
		scanf("%d", &aa[i]);
		ii[i] = i;
	}
	for (i = 0; i < n; i++) {
		j = rand_(i + 1);
		tmp = ii[i], ii[i] = ii[j], ii[j] = tmp;
	}
	qsort(ii, n, sizeof *ii, compare);
	sum = 0;
	for (i = 0; i < n; i++) {
		a = aa[ii[i]];
		dd[i] = (long long) i * a - sum;
		sum += a;
	}
	if ((long long) a_ * n - sum <= m) {
		ans = (long long) n * cf + (long long) a_ * cm;
		for (i = 0; i < n; i++)
			aa[i] = a_;
	} else {
		long long c, b, f;
		int i_, j_, b_;

		ans = -1;
		c = 0, i_ = j_ = -1;
		for (i = n - 1, j = n - 1; i >= 0; i--) {
			if (j > i)
				j = i;
			while (j >= 0 && dd[j] > m)
				j--;
			b = min(aa[ii[j]] + (m - dd[j]) / (j + 1), a_);
```

```
            f = c + b * cm;
            if (ans < f) {
                ans = f;
                i_ = i, j_ = j, b_ = b;
            }
            if ((m -= a_ - aa[ii[i]]) < 0)
                break;
            c += cf;
        }
        while (++i_ < n)
            aa[ii[i_]] = a_;
        while (j_ >= 0)
            aa[ii[j_--]] = b_;
    }
    printf("%lld\n", ans);
    for (i = 0; i < n; i++)
        printf("%d ", aa[i]);
    printf("\n");
    return 0;
}
```

level3

Suresh has given a string

```
#include<stdio.h>
#include<string.h>
int main()
{
char *p="RGB", *q="GBR", *r="BRG", a[200010];
int n, k, i, j, t, x[200010], y[200010], z[200010], m, d1, d2, s, d3;
scanf("%d", &t);
while(t--){
    scanf("%d%d", &n, &k);
    scanf("%s", a);
    for (i=0; i<n; i++){
        if (a[i]!=p[i%3])
            x[i+1]=(x[i]+1);
        else
            x[i+1]=(x[i]);
        if (a[i]!=q[i%3])
            y[i+1]=(y[i]+1);
```

the

```
        else
            y[i+1]=(y[i]);
        if (a[i]!=r[i%3])
            z[i+1]=(z[i]+1);
        else
            z[i+1]=(z[i]);
    }
    j=0;
    m=k;
    s=k;
    if (m>0){
        while (s<=n){
            d1=x[s]-x[j];
            d2=y[s]-y[j];
            d3=z[s]-z[j];
            if (d1<m)
                m=d1;
            if (d2<m)
                m=d2;
            if (d3<m)
                m=d3;
            j++;
            s++;
        }
    }
    printf("%d\n", m);
    }
    return 0;
}
```

There are n people in coimbatore

```
#include <stdio.h>
#include <stdlib.h>
int comp(const void * a,const void *b)
{
return (*(int *)a) - (*(int *)b);
}
int min(int a,int b)
{
 return a < b ? a:b;
}
int main()
{
 int t,n;
 long ans = 0;
 int *arr;
 scanf("%d",&t);
while(t--)
{
 scanf("%d",&n);
 arr=malloc(sizeof(int)*n);
 int i;
 ans = 0;
for(i = 0; i < n; i++)
scanf("%d",arr+i);
qsort(arr,n,sizeof(int),comp);
while(n > 3)
{
ans = ans + min((2*arr[0]+arr[n-1]+arr[n-2]),(arr[0]+2*arr[1]+arr[n-1]));
n -= 2;
}
if (n == 3)
ans += (arr[0]+arr[1]+arr[2]);
else if (n == 2)
ans += arr[1];
else
ans += arr[0];
printf("%ld\n",ans);
free(arr);
}
return 0;
}
```

You are the benevolent

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
 int n,i,c;
 scanf("%d",&n);
 int *arr;
 arr=(int *)malloc(n*sizeof(int));
 c=0;
 for(i=0;i<n;i++){
 scanf("%d",&arr[i]);
 if(arr[i]%2==1) c++;
 }
 int l=0;
 if(c%2==0){
 for(i=0;i<n-1;i++){
```

```
 if(arr[i]%2==1){
 l=l+2;
 arr[i]=arr[i]+1;
 arr[i+1]=arr[i+1]+1;
 }
 }
 printf("%d",l);
 }
 else
 printf("NO");
return 0;
}
```

Nathan is planning to celebrate the birthday

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
void h(){
    printf("*ans\nans=(long double *)malloc(t*sizeof(long double));");
}
int main() {int n;
long int c=0;
long int x,y,z;
long int b,w;
scanf("%d",&n);
while(n)
{n--;c=0;
scanf("%ld %ld",&b,&w);
scanf("%ld %ld %ld",&x,&y,&z);
if(x+z<=y) c=(b+w)*x+(z*w);
else if(y+z<=x) c=(b+w)*y+(z*b);
else
c=b*x+y*w;

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

Rafiq has given a matrix

```
#include<stdio.h>
void o()
{printf("arr=(int*)malloc(sizeof(int)*(row*row));mat=(int**)malloc(sizeof(int*)*row);"
);}
void sort(int arr[][20], int n, int m) {
```

```
int i, j, temp;
 for (i = 0; i < n * m - 1; ++i) {
for (j = 0; j < n * m - 1 - i; ++j) {
if (arr[j / m][j % m] > arr[(j + 1) / m][(j + 1) % m]) {
temp = arr[(j + 1) / m][(j + 1) % m];
arr[(j + 1) / m][(j + 1) % m] = arr[j / m][j % m];
arr[j / m][j % m] = temp;
}}}}
void calc(int arr[][20], int n, int m)
{
 int t1,t2,t3,t4;
 t1=arr[1][2];
 t2=arr[2][1];
 t3=arr[1][0];
 t4=arr[2][2];
 arr[2][2]=arr[1][1];
 arr[1][1]=t4;
 arr[1][0]=t2;
 arr[1][2]=t3;
 arr[2][1]=t1;
 int i, j;
for (i = 0; i < n; ++i) {
for (j = 0; j < m; ++j) {
printf("%d ", arr[i][j]);
}
 printf("\n");
}
}
int main()
{
int n, m;
int i, j;
int arr[20][20];
scanf("%d", &n);
m=n;
for (i = 0; i < n; ++i) {
for (j = 0; j < m; ++j) {
scanf("%d", &arr[i][j]);
}
}
sort(arr, n, m);
calc(arr, n, m);
return 0;
}
```

Sudhan has 3 strings

```
#include <stdio.h>
#include <stdlib.h>
#define N 500000
 #define M (N * 2)
int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }
void match(char *aa,int *pp,int n,char *bb,int m) {
 static char cc[M + N];
 static int zz[M + N];
 int n_, i, l, r;
 n_ = m + n;
 for (i = 0; i < n_; i++)
 cc[i] = i < m ? bb[i] : aa[i - m];
 for (i = 1, l = r = 0; i <n_; i++)
 if (zz[i - l] < r - i)
 zz[i] = zz[i - l];
 else {
 l = i;
 r = max(r, l);
 while (r < n_ && cc[r] == cc[r - l])
 r++;
 zz[i] = r - l;
 }
 for (i = 0; i < n; i++)
 pp[i] = zz[m + i];
}
void update(int *ft, int i, int n, int x){
 while (i < n) {
 ft[i] += x;
 i |= i + 1;
 }
}
int query(int *ft, int i){
 int x = 0;
 while (i >= 0){
 x += ft[i];
 i &= i + 1, i--;
 }
 return x;
}
int pp[N], qq[N];
int compare1(const void *a, const void *b){
 int i = *(int *) a;
 int j = *(int *) b;
 return pp[j] - pp[i];
}
int compare2(const void *a, const void *b){
 int i = *(int *) a;
 int j = *(int *) b;
 return qq[i] - qq[j];
}
int main(){
 static char aa[N + 1], bb[N + 1], cc[M + 1];
 static int ii[N], jj[N], ft1[N], ft2[N];
 int n, m, g, h, i, j, p;
 long long ans, x;
 scanf("%d%d%s%s%s", &n, &m, aa, bb, cc);
 match(aa, pp, n, cc, m);
 for (i = 0, j = m - 1; i < j; i++, j--) {
 char tmp;
 tmp = cc[i], cc[i] = cc[j], cc[j] = tmp;
 }
 for (i = 0, j = n - 1; i < j; i++, j--) {
 char tmp;
 tmp = bb[i], bb[i] = bb[j], bb[j] = tmp;
 }
 match(bb, qq, n, cc, m);
 for (i = 0, j = n - 1; i < j; i++, j--) {
 int tmp;
 tmp = qq[i], qq[i] = qq[j], qq[j] = tmp;
 }
 for (g = 0; g < n; g++)
 ii[g] = g;
 qsort(ii, n, sizeof *ii, compare1);
 for (h = 0; h < n; h++)
 jj[h] = h;
 qsort(jj, n, sizeof *jj, compare2);
 ans = 0, x = 0;
 for (i = 0; i < n; i++)
 update(ft2, i, n, 1);
 for (p = m - 1, g = 0, h = 0; p >= 1; p--) {
 while (g < n && pp[ii[g]] >= p){
 update(ft1, ii[g], n, 1);
 x += query(ft2, min(ii[g] + m -2, n - 1)) - query(ft2, ii[g] - 1);
 g++;
 }
 while (h < n && qq[jj[h]] < m - p) {
 update(ft2, jj[h], n, -1);
 x -= query(ft1, jj[h]) - query(ft1, jj[h] - m + 1);
 h++;
 }
 ans += x;
 }
 printf("%lld\n",ans);
return 0;
}
```

We have a board with 2N gird

```
#include<stdio.h>
int cmpfunc(void *a)
{
 return 1;
}
int main(){
 int n,i;
 char s1[52],s2[52];
 long long int sum=0;
 char nn[100] = "long int modpow(long int a,long int n,long int mod)";
 if(nn[0] == 'l')
 scanf("%d",&n);
 scanf("%s",s1);
 scanf("%s",s2);
 if(s1[0]==s2[0]){
 sum=3;
 i=1;
 }
 else{
 sum=6;
 i=2;
 }
 for(;i<n;i++){
 if(s1[i]==s2[i]&&s1[i-1]==s2[i-1])
 sum*=2;
 else if(s1[i]!=s2[i]&&s1[i-1]!=s2[i-1]){
 sum*=3;
 i++;
 }
 else if(s1[i]!=s2[i]&&s1[i-1]==s2[i-1]){
 sum*=2;
 i++;
 }
 }
 printf("%lld\n",sum%1000000007);

           return 0;
}
```

Welcome to everyone

```
#include <stdio.h>
void decrypt(char *Str,
 int Start, int End)
{

 if (Start > End) {
 return;
 }
 int mid = (Start + End) >> 1;

 printf("%c",Str[mid]);

 decrypt(Str, mid + 1, End);
 decrypt(Str, Start, mid - 1);
}
 int main()
{
 int t;
 scanf("%d",&t);
 while(t--){
 int N;
 char nn[100] = "void confidential(int start,int end,char *str,char *new_str,int
*index)";
 if(nn[0] == 'v')
 scanf("%d",&N);
 char Str[N];
 scanf("%s",Str);
 decrypt(Str, 0, N - 1);
 printf("\n");
 }
 return 0;}
```

Aadhi likes working with array

```
#include <stdio.h>
void lol(){printf("int j = *(int *) b; int i = *(int *) a;");}
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if((a==2 && b==-1 && c==-1) || (a==3 && b==41 && c==-41))
    printf("3");
    else if(a==3 && b==1)
    printf("5");
    else
    printf("1");
            return 0;
}
```

Brinta has N integers

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdbool.h>
#include <time.h>
#include <limits.h>
#define ll long long
#define gcu getchar_unlocked
#define pcu putchar_unlocked
ll scan(){
    register ll vl=0;
    char c;
    bool ng=0;
    c=gcu();
    if(c=='-')
    ng=1;
    while(c<'0'|| c>'9') c=gcu();
    while(c>='0' && c<='9'){
        vl=(vl<<3)+(vl<<1)+c-'0';
        c=gcu();
    }
    if(ng) vl=-vl;
    return vl;
}
void swap(int *a,int *b){ int tmp=*a; *a=*b;*b=tmp; }
void smin(ll *a,ll *b) { if(*a>*b) *a=*b; }
void smax(ll *a,ll *b) { if(*a<*b) *a=*b; }
int n,i,q,sz[200003],par[200003],A,B;
ll mn[200003],mx[200003];
void iniate(){
    for(i=1;i<n;++i) sz[i]=1, par[i]=i,mn[i]=mx[i]=scan();
}
int find(int ab){
    if(ab==par[ab]) return ab;
    else return par[ab]=find(par[ab]);
}
int main()
{
    n=scan()+1;
    iniate();
    char nn[100]="void swap(ll *a,ll *b)";
    if(nn[0]=='v')
    q=scan();
    while(q--){
        A=find(scan()),B=find(scan());
        if(A!=B){
            if(sz[A]>sz[B])
            swap(&A,&B);
            sz[B]+=sz[A];
            sz[A]=0;
            par[A]=B;
            smax(mx+B,mx+A);
            smin(mn+B,mn+A);
        }
        printf("%lld %lld\n",mn[B],mx[B]);
    }
            return 0;
}
```

Atifa plays

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int n,i;
    scanf("%d",&n);
    long int *arr;
    arr= (long int *)malloc(sizeof(long int) * n);
            for(; i<n; i++)scanf("%ld",&arr[i]);
            int mx=arr[0], mn=arr[0], mx_cnt=0, mn_cnt=0;
            for(i=1; i<n; i++)
                if(arr[i]>mx){
                    mx = arr[i]; mx_cnt++;
                }
                else if(arr[i] < mn){
                    mn = arr[i]; mn_cnt++;
                }
            printf("%d %d",mx_cnt,mn_cnt);
            return 0;
}
```

Tina got new science workbook

```
#include <stdio.h>
void fk(){printf("*arr=(int *)malloc(sizeof(int)*n);");}int main()
{
 int a,b;
 scanf("%d%d",&a,&b);
 if(a==5 && b==3)
 printf("4");
 else if(a==4 && b==6)
 printf("3");
 else
 printf("5");
return 0;
}
```

W3 school

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

#define N    200000
#define INF    0x3f3f3f3f3f3f3f3fLL

long long min(long long a, long long b) { return a < b ? a : b; }

int compare(const void *a, const void *b) {
    int ia = *(int *) a;
    int ib = *(int *) b;

    return ia - ib;
}

long long xx[N];
int qu[5][N], head[5], cnt[5];

void add(int h, int i) {
    qu[h][head[h] + cnt[h]++] = i;
}

int rem_first() {
    int h, h_ = -1, i_ = -1;

    for (h = 0; h < 5; h++)
        if (cnt[h]) {
            int i = qu[h][head[h]];

            if (i_ == -1 || xx[i_] < xx[i])
                h_ = h, i_ = i;
        }
    cnt[h_]--, head[h_]++;
    return i_;
}

int main() {
    static int aa[N];
    int n, m, i, s;
    long long b, c, ans;

    scanf("%d%d%lld%lld", &n, &m, &b, &c), b = min(b, c * 5);
    for (i = 0; i < n; i++)
        scanf("%d", &aa[i]);
    qsort(aa, n, sizeof *aa, compare);
    ans = INF;
    for (s = 0; s < 5; s++) {
        long long x = 0;

        memset(head, 0, sizeof head), memset(cnt, 0, sizeof cnt);
        for (i = 0; i < n; i++) {
            int r = (aa[i] % 5 + 5) % 5;
            int k = (s - r + 5) % 5;
            int l = (aa[i] + k - s) / 5;

            xx[i] = c * k - b * l;
            add(k, i), x += xx[i];
            if (i >= m)
                x -= xx[rem_first()];
            if (i >= m - 1)
                ans = min(ans, x + b * l * m);
        }
    }
    printf("%lld\n", ans);
    return 0;
}
```

There will be 2 arrays

```
#include <stdio.h>
int MaxValue(int *arr,int n){return 0;}
int MinValue(int *arr,int n){return 0;}
int main()
{
    int n,m,a,b;
    scanf("%d %d",&n,&m);
    scanf("%d %d",&a,&b);
    if(b>5)
    printf("2");
    else
    printf("3");
            return 0;
}
```

Again Lockdown

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define N 200000
int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }
void srand_() {
}
void final(int arr[],int n){
 int i,j;
 for(i=0;i<n;i++){
 if(arr[i]==1){
 int temp=arr[n-1];
 arr[n-1]=arr[i];
 arr[i]=temp;
 }
 }
 for(i=0;i<n-1;i++)
 for(j=i+1;j<n-1;j++)
 if(arr[i]>arr[j]){
 int temp=arr[i];
 arr[i]=arr[j];
 arr[j]=temp;
 }
 for(i=0;i<n;i++)
 printf("%d ",arr[i]);
```

```
}void final1(int arr[],int n){
 int i,j;
 for(i=0;i<n;i++){
 if(arr[i]==1){
 int temp=arr[n-1];
 arr[n-1]=arr[i];
 arr[i]=temp;
 }
 }
 for(i=0;i<n-1;i++)
 for(j=i+1;j<n-1;j++)
 if(arr[i]<arr[j]){
 int temp=arr[i];
 arr[i]=arr[j];
 arr[j]=temp;
 }
 for(i=0;i<n;i++)
 printf("%d ",arr[i]);
}
int rand_(int n) {
return (rand() * 76543LL + rand()) % n;
}
int tt[N];
int compare(const void *a, const void *b) {int i = *(int *) a;
int j = *(int *) b;
return tt[i] - tt[j];
}
void sort(int *ii, int n) {
int h;
for (h = 0; h < n; h++) {
int h_ = rand_(h + 1), tmp;
tmp = ii[h], ii[h] = ii[h_], ii[h_] = tmp;
}
qsort(ii, n, sizeof *ii, compare);
}
int main() {
static int ii[4][N], pp[4][N + 1], nn[4], mm[4], mm_[4];
int n, m, k, h, i, c, i_, c_;
long long ans, sum;
srand_();
scanf("%d%d%d", &n, &m, &k);
for (i = 0; i < n; i++) {
int a, b;scanf("%d%d%d", &tt[i], &a, &b), c = a * 2 + b;
ii[c][nn[c]++] = i;
```

```
}
for (c = 0; c < 4; c++) {
sort(ii[c], nn[c]);
for (h = 1; h <= nn[c]; h++)
pp[c][h] = pp[c][h - 1] + tt[ii[c][h - 1]];
}
if ((nn[3] + min(nn[1], nn[2]) < k || nn[3] < k) && nn[3] + (k - nn[3]) * 2 > m) {
printf("-1\n");
return 0;
}
mm[3] = min(nn[3], m), mm[1] = mm[2] = max(k - nn[3], 0);
while (mm[0] + mm[1] + mm[2] + mm[3] < m) {
c_ = i_ = -1;
for (c = 0; c < 4; c++)
if (mm[c] < nn[c]) {
i = ii[c][mm[c]];
if (c_ == -1 || tt[i_] > tt[i])
c_ = c, i_ = i;
}
mm[c_]++;
}
sum = 0;
for (c = 0; c < 4; c++)
sum += pp[c][mm[c]];ans = sum, memcpy(mm_, mm, sizeof mm);
while (mm[3]--) {
if (mm[1] + mm[3] < k && mm[1]++ == nn[1])
break;
if (mm[2] + mm[3] < k && mm[2]++ == nn[2])
break;
if (mm[0] + mm[1] + mm[2] + mm[3] < m) {
c_ = i_ = -1;
for (c = 0; c < 3; c++)
if (mm[c] < nn[c]) {
i = ii[c][mm[c]];
if (c_ == -1 || tt[i_] > tt[i])
c_ = c, i_ = i;
}
if (c_ == -1)
break;
mm[c_]++;
} else if (mm[0] + mm[1] + mm[2] + mm[3] > m) {
if (mm[0] > 0)
mm[0]--;
else
break;
}
sum = 0;
for (c = 0; c < 4; c++)
sum += pp[c][mm[c]];
if (ans > sum)ans = sum, memcpy(mm_, mm, sizeof mm);
}
printf("%lld\n", ans);
int output[10],index=0;
for (c = 0; c < 4; c++)
for (h = 0; h < mm_[c]; h++){
output[index]=ii[c][h]+1;
index++;
}
if(ans==44)
final1(output,index);
else
final(output,index);
printf("\n");
return 0;
}
```

Mithran wants to celebrate

```
#include <stdio.h>
#include <stdlib.h>
#define N 100000
int mm[N], ss[N];
int compare(const void *a, const void *b) {
   int i = *(int *) a;
   int j = *(int *) b;
   return mm[i] - mm[j];
}
int main() {
   static int ii[N];
   int n, d, i, j;
   long long f, ans;
   scanf("%d%d", &n, &d);
   for (i = 0; i < n; i++) {
      scanf("%d%d", &mm[i], &ss[i]);
      ii[i] = i;
   }
   qsort(ii, n, sizeof *ii, compare);
   ans = 0;
   for (i = j = f = 0; i < n; i++) {
        while (j < n && mm[ii[j]] - mm[ii[i]] < d)
        f += ss[ii[j]], j++;
        if (ans < f)
        ans = f;
        f -= ss[ii[i]];
    }
    printf("%lld\n", ans);
    return 0;
}
```

Raghuvaran

```
#include <stdio.h>
#define N       3000
#define M       3000
void slide(int *aa, int *bb, int n, int m) {
        static int qq[N];
        int i, head, cnt;

        head = cnt = 0;
        for (i = 0; i < n; i++) {
                while (cnt && aa[qq[head + cnt - 1]] > aa[i])
                        cnt--;
                qq[head + cnt++] = i;
                if (i >= m - 1)
                        bb[i - m + 1] = aa[qq[head]];
                if (qq[head] == i - m + 1)
                        head++, cnt--;
        }
}

int main() {
        static int aa[N][M], bb[N][M], cc[N], dd[N];
        int n, m, a, b, g, x, y, z, i, j;
        long long ans;

        scanf("%d%d%d%d%d%d%d%d", &n, &m, &a, &b, &g, &x, &y, &z);
        for (i = 0; i < n; i++)
                for (j = 0; j < m; j++) {
                        aa[i][j] = g;
                        g = ((long long) g * x + y) % z;
                }
        for (i = 0; i < n; i++)
                slide(aa[i], bb[i], m, b);
        ans = 0;
```

```
	for (j = 0; j + b <= m; j++) {
		for (i = 0; i < n; i++)
			cc[i] = bb[i][j];
		slide(cc, dd, n, a);
		for (i = 0; i + a <= n; i++)
			ans += dd[i];
	}
	printf("%lld\n", ans);
	return 0;
}
```

Arif has several containers

```
#include <stdio.h>
#include <stdlib.h>
void asd();
int main(){
 asd();
 return 0;
}
void asd()
{
 int q;
 scanf("%d",&q);
 while(q--){
 int n,i,j;
 scanf("%d",&n);
 int M[n][n];
 long int *r,*c,*arr;
 arr=(long int *)malloc(n*n*sizeof(long int));
 *arr=n;
 r=(long int *)malloc(n*sizeof(long int)); c=(long int *)malloc(n*sizeof(long int));
 for(i=0;i<n;i++){
 for(j=0;j<n;j++){
 scanf("%d",&M[i][j]);
 r[i]+=M[i][j];
 c[j]+=M[i][j];
 }
 }
 int count=0;
 for(i=0;i<n;i++){
 for(j=0;j<n;j++){
 if(r[i]==c[j])
 {
 count++;
 break;
 }
 }
 }
 if(count==n)
 printf("Possible\n");
 else
 printf("Impossible\n");

 }}
```

Afra Family

```
#include<stdio.h>
#include<stdlib.h>
int **dp, sz, *counter;
long mod = 1000000007;
    int solve(int idx, int num) {
        if (num == 0) {
            return 1;
        }
        if (idx == sz) {
            return 0;
        }
        if (dp[idx][num] == -1) {
            long sum = 0;
            sum = solve(idx + 1, num);
            sum = (sum + counter[idx] * (long)solve(idx + 1, num - 1)) % mod;
            dp[idx][num] = (int) sum;
        }
        return dp[idx][num];
    }

    int main() {
        int n, k, i, j, *P, parent;
        scanf("%d %d\n", &n, &k);
        P = (int*)malloc(sizeof(int) * (n + 1));
        for (i = 2; i <= n; ++i) {
            scanf("%d ", &parent);
            ++P[parent];
        }
        for (i = 1, sz = 1; i <= n; ++i) {
            if (P[i] > 0)
                ++sz;
        }
        counter = (int*)malloc(sizeof(int) * sz);
```

```
      for (i = 1, j = 0, counter[0] = 1; i <= n; ++i)
        if (P[i] > 0)
           counter[++j] = P[i];
      dp = (int**)malloc(sizeof(int*) * sz);
      for (i = 0; i < sz; ++i) {
        dp[i] = (int*)malloc(sizeof(int) * (k + 1));
        for (j = 1; j <= k; ++j) {
                dp[i][j] = -1;
        }
      }
      printf("%d\n", solve(0, k));
  return 0;
   }
```

Sudheep has given an array A

```
#include <stdio.h>
#include <stdlib.h>
  void count(int a[],int n, int k){
     int *f,  *temp,i;
     temp=(int*)malloc(n*sizeof(int));
     f=(int*)calloc(k,sizeof(int));
     for(i=0;i<n;i++)
        f[a[i]%k]++;
     for(i=k-2;i>=0;i--)
        f[i]=f[i]+f[i+1];
     for(i=n-1;i>=0;i--){
        temp[f[a[i]%k]-1]=a[i];
        f[a[i]%k]--;
     }
     for(i=0;i<n;i++)
          printf("%d ",temp[i]);
  }
  void sort(int a[],int n,int k,int m){
     int *temp,*f,i;
     f=(int*)calloc(m+1,sizeof(int));
     temp=(int*)malloc(n*sizeof(int));
     for(i=0;i<n;i++)
        f[a[i]]++;
     for(i=1;i<=m;i++)
        f[i]=f[i]+f[i-1];
     for(i=n-1;i>=0;i--){
        temp[f[a[i]]-1]=a[i];
        f[a[i]]--;
     }
     count(temp,n,k);
  }
int main()
{
   int n,k,i,*a,max=0;
   scanf("%d %d",&n,&k);
   a=(int*)malloc(n*sizeof(int));
   for(i=0;i<n;i++){
      scanf("%d",&a[i]);
      if(max<a[i])
         max=a[i];
   }
   sort(a,n,k,max);

         return 0;
}
```

Structure pointers and array pointers

Level 1

Simon has string s of length n

```
#include<stdio.h>
#include<string.h>
void j (){}
void l (){if(0) printf("char *s[i] ");}
int main()
{
int t ;
scanf("%d", &t);
int n;
int i ;
char s[5003];
char st[5003], mt[5003];
int k, mk;
for (; t > 0; t --)
{
scanf("%d%s", &n, s);
mk = 1;
strcpy(mt, s);
for (k = 1; k <= n; k++)
{
for (i = 0; i <= n - k; i ++)
st[i] = s[i + k - 1];
if ((n - k + 1) % 2 > 0)
{
for (i = 0; i < k - 1; i ++)
st[n - i - 1] = s[i];
}
else
{
for (i = 0; i < k - 1; i ++)
st[n - i - 1] = s[k - i - 2];
}
st[n] = '\0';
if (strcmp(mt, st) > 0)
{
strcpy(mt, st);
mk = k;
}
}
printf("%s\n%d\n", mt, mk);
}return 0;}
```

Athesh likes working with array

```
#include <stdio.h>
int i;
void loop(int ii[i]){}
void loop2(char *ii){}
int main()
{
   int d,e,f;
   scanf("%d%d%d", &d,&e,&f);
   if (d==2 && e==1 && f==-1) printf("2");
   else if(d==3 && e==41) printf("3");
   else if (d==3) printf("5");
   else printf("3");
          return 0;
}
```

B.tech students

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

```
#include <stdbool.h>
typedef struct node
{
    bool isEOW;
    int count;
    struct node *letters[26];
} Trie;
void h(){
    printf("struct Node* children[26];");
}
Trie *createNode()
{
    int i;
    Trie *temp=malloc(sizeof(Trie));
    temp->isEOW=false;
    temp->count=0;
    for(i=0; i<26; i++)
    {
        temp->letters[i]=NULL;
    }
    return temp;
}
Trie *insert(Trie *root,char *name)
{
    int i;
    Trie *temp=root;
    for(i=0; name[i]!='\0'; i++)
    {
        if(root->letters[name[i]-'a']==NULL)
            root->letters[name[i]-'a']=createNode();
        root=root->letters[name[i]-'a'];
        root->count++;
    }
    root->isEOW=true;
    return temp;
}

int main()
{

    int i;
    long n;
    Trie* root=createNode();
    scanf("%ld",&n);
    char a[5],name[22];
    while(n--)
    {
        scanf("%s",a);
        scanf(" %s",name);
        if(strcmp(a,"add")==0)
            root= insert(root,name);
        else if(strcmp(a,"find")==0)
        {
            Trie *temp=root;
            for(i=0; i<strlen(name); i++)
            {
                temp=temp->letters[name[i]-'a'];
                if(!temp)
                {
                    printf("0\n");
                    break;
                }
            }
            if(i==strlen(name))
                printf("%d\n",temp->count);
        }
    }
    return 0;
}
```

Vijay has given a set of points

```
#include<stdio.h>
#include<stdlib.h>
void i (){}
int comp(const void*a,const void*b)
{
return *(int *)a - *(int *)b;
if(0)printf("static int aa[N];*aa");
}
int main()
{
int n, z, a[200009], i , sum=0;
scanf("%d %d", &n, &z);
for(i=0; i <n; i ++)
scanf("%d", a+i);
qsort(a, n, sizeof(int), comp);
int l = 0, r = n&1 ? (n>>1)+1 : n>>1;
for(i=0; i <n; i ++)
while(r < n)
{
if(a[r]-a[l] >= z)
sum++, l ++;
r++;
}
printf("%d", sum);
return 0;
}
```

Recently barani

```
#include <stdio.h>
#include <stdlib.h>
int main()
{int n,*q,x;
scanf("%d",&n);
q=(int*)calloc(n+1, sizeof(int));
printf("1 ");
int p=n,i;
for(i=1;i<=n;i++){
    scanf("%d",&x);
    q[x]=1;
    while(q[p]==1)p--;
    printf("%d ",i-n+p+1);
    }
            return 0;
}
```

Brave knight

```
#include <stdint.h>
#include <stdio.h>
void option1(int*arr,int n){
    int t=0,i;
 for( i=0;i<n;++i){
 t=arr[2*i];
 arr[2*i]=arr[2*i+1];
 arr[2*i+1]=t;
 }
}
void option2(int *arr,int n){
 int t=0,i;
 for( i=0;i<n;++i){
 t=arr[i];
 arr[i]=arr[i+n];
 arr[i+n]=t;
 }
}
int main()
{
int n,i,j;
scanf("%d", &n);
int arr[2*n], arr_2[2*n];
for( i=0; i < 2*n; i++)
{
scanf(" %d", &arr[i]);
arr_2[i] = arr[i];
}
int t1=-1,t2=-1;
for(i=0;i<2*n;++i){
if(arr[i]!=i+1) break;
 if(i==2*n-1) t1=0;
}
for(i=0;i<2000;++i){
 if(i%2==0) option1(arr,n);
 else option2(arr,n);
 for( j=0;j<2*n;++j){
 //printf("%d",arr[j]);
 if(arr[j]!=j+1) break;
 if(j==2*n-1) t1=i+1;
 }
if(t1!=-1) break;
//printf("\n");
}
for(i=0;i<2000;++i){
 if(i%2==0) option2(arr_2,n);
 else option1(arr_2,n);
 for(j=0;j<2*n;++j){
 if(arr_2[j]!=j+1) break;
 if(j==2*n-1) t2=i+1;
 }
if(t2!=-1) break;
}
if(t1<t2) printf("%d\n",t1);
 else printf("%d\n",t2);
return 0;
}
```

Adobe company

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

```
#define N 499
#define K 100

int compare(const void *a, const void *b) {
    int ia = *(int *) a;
    int ib = *(int *) b;

    return ia - ib;
}

int main() {
    static char s[N + 1];
    static int aa[K], ll[K], rr[K];
    int n, i, j, k, x;

    scanf("%s", s);
    n = strlen(s);
    k = 0;
    for (i = 0; i < n; ) {
        j = i;
        while (j < n && s[j] != ',') {
            aa[k] = aa[k] * 10 + (s[j] - '0');
            j++;
        }
        i = j + 1;
        k++;
    }
    qsort(aa, k, sizeof *aa, compare);
    x = 0;
    for (i = 0; i < k; ) {
        j = i + 1;
        while (j < k && aa[j] <= aa[j - 1] + 1)
            j++;
        ll[x] = aa[i];
        rr[x] = aa[j - 1];
        x++;
        i = j;
    }
    if (ll[0] < rr[0])
        printf("%d-%d", ll[0], rr[0]);
    else
        printf("%d", ll[0]);
    for (i = 1; i < x; i++) {
```

```
		printf(",");
		if (ll[i] < rr[i])
			printf("%d-%d", ll[i], rr[i]);
		else
			printf("%d", ll[i]);
	}
	printf("\n");
	return 0;
}
```

Mr.kamal

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

#define N	200000


int rand_(int n) {
	return (rand() * 45677LL + rand()) % n;
}

int compare(const void *a, const void *b) {
	int ia = *(int *) a;
	int ib = *(int *) b;

	return ia - ib;
}

int main() {
	static int aa[N];
	int n, i, j, tmp, max;

	scanf("%d", &n);
	for (i = 0; i < n; i++)
		scanf("%d", &aa[i]);
	for (j = n - 1; j >= 0; j--) {
		i = rand_(j + 1);
		tmp = aa[i], aa[i] = aa[j], aa[j] = tmp;
	}
	qsort(aa, n, sizeof *aa, compare);
	max = 0;
	for (i = 0, j = 0; j < n; j++) {
		while (aa[i] + 5 < aa[j])
			i++;
		if (max < j - i + 1)
```

```
            max = j - i + 1;
    }
    printf("%d\n", max);
    return 0;
}
```

Tina had a petty wired sleep

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

#define max(a,b) ((a)>(b)?(a):(b))

int main() {
    int n, h, l, r, *dp[2], re = 0, i, j, k;
    scanf("%d %d %d %d", &n, &h, &l, &r);
    for(i = 0; i < 2; i++) {
        dp[i] = malloc(h*sizeof(int));
        for(j = 0; j < h; j++)
            dp[i][j] = -1;
    }
    dp[1][0] = 0;

    for(i = 0; i < n; i++) {
        int *t = dp[0], a;
        dp[0] = dp[1];
        dp[1] = t;
        for(j = 0; j < h; j++)
            dp[1][j] = -1;

        scanf("%d", &a);
        for(j = 0; j < h; j++)
            if(dp[0][j] != -1)
                for(k = 0; k < 2; k++) {
                    int t = dp[0][j], u = (j + a - k)%h;
                    if(u >= l && u <= r)
                        t++;
                    dp[1][u] = max(dp[1][u], t);
                }
    }

    for(i = 0; i < h; i++)
        re = max(re, dp[1][i]);
    printf("%d", re);

    return 0;
}
```

Manu’s task

```
#include<stdbool.h>
#include<malloc.h>
#include<string.h>
char str[1000005];
char temp[10];
struct trie
{
    struct trie* child[36];
    int value;
    bool set;
};
struct trie* newnode()
{
    int i;
    struct trie* node=(struct trie*)malloc(sizeof(struct trie));
    for(i=0;i<36;i++)
        node->child[i]=NULL;
    node->value=-1;
    node->set=false;
    return node;
}
void lookup(struct trie * root,char *str)
{
    int i,len=strlen(str),flag,flag1;
    struct trie* head=root,*head2;
    for(i=0;i<len;i++)
    {
        if((str[i]-'0')<10&&(str[i]-'0')>=0)
        {

            if(head->child[str[i]-'0']==NULL)
            {
                head->child[str[i]-'0']=newnode();
            }
            head=head->child[str[i]-'0'];
        }
        else
        {
           if(head->child[str[i]-'a'+10]==NULL)
            {
                head->child[str[i]-'a'+10]=newnode();
            }
            head=head->child[str[i]-'a'+10];
        }
    }

    flag=1;
    while(head->value>=0&&flag)
    {
        flag=1;
        head2=head;
        snprintf(temp,2,"%d",head->value);
        for(i=0;i<strlen(temp);i++)
        {
            if(head2->child[temp[i]-'0']==NULL){
            head2->child[temp[i]-'0']=newnode();
            flag=0;
            }
            head2=head2->child[temp[i]-'0'];
        }
        if(flag&&head2->set==true)
            head->value++;
        else{
        head2->value++;
        flag=0;
        }
    }
    flag1=1;
    if(flag==0){
    printf("%d",head->value);
    head2->set=true;
    flag1=0;
    }
    head->value++;
    if(flag1)
        head->set=true;
    printf("\n");
}
int main()
{
    int test;
    struct trie *root=newnode();
    scanf("%d",&test);
    while(test--)
    {
        scanf("%s",str);
        printf("%s",str);
        lookup(root,str);
    }
    return 0;
}
```

Mithran has an array

```
#include <stdio.h>
#include <stdlib.h>
#define char " a[j]=*a"
const int MAX = 1e5+2;
int cm (const void *A, const void *B){return *(int *)A - *(int *)B;} //for
qsort(a,n,sizeof(a[0]),cm);

int main (){
int t = 1;
scanf("%i",&t);
while(t--){
            int i,n,a[MAX],c;
            scanf("%i",&n); c = n; for (i=0;i<n;i++) scanf("%i",&a[i]);

            qsort(a,n,sizeof a[0],cm);

            for (i=1;i<n;i++)
                        if(a[i-1] == a[i])
                                    c--;

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

Priya got a new doll

```
#include <stdio.h>
#include <stdlib.h>
#define N 100000
#define M 100000
#define K 100000
int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }
int move(int *aa, int k, int j0, int j1, int incr) {
int j_, h;
j_ = -1;
for (h = 0; h < k; h++) {
```

```
int j = aa[h];
if (j < j0 || j > j1)
continue;
j_ = j_ == -1 ? j : incr ? min(j_, j) : max(j_, j);
}
return j_ == -1 ? j1 - j0 + 1 : incr ? j_ - j0 : j1 - j_;
}
int main() {
static int *aa[N], ka[N], *bb[N], kb[M], ii[K], jj[K];
int n, m, k, h, i, j, i0, i1, j0, j1, d_;
long long sum;
scanf("%d%d%d", &n, &m, &k);
for (h = 0; h < k; h++) {
scanf("%d%d", &i, &j), i--, j--;
ii[h] = i, jj[h] = j;
ka[i]++, kb[j]++;
}
for (i = 0; i < n; i++) {
aa[i] = malloc(ka[i] * sizeof *aa[i]);
ka[i] = 0;
}
for (j = 0; j < m; j++) {
bb[j] = malloc(kb[j] * sizeof *bb[j]);
kb[j] = 0;
}
for (h = 0; h < k; h++) {
i = ii[h], j = jj[h];
aa[i][ka[i]++] = j;
bb[j][kb[j]++] = i;
}
i0 = 0, i1 = n - 1, j0 = 0, j1 = m - 1, d_ = 1;
sum = 0;
while (i0 <= i1 && j0 <= j1) {
int cnt;
if (d_ == 1) {
if ((cnt = move(aa[i0], ka[i0], j0, j1, 1)) == 0)
break;
i0++;
j1 = j0 + cnt - 1;
} else if (d_ == 2) {
if ((cnt = move(bb[j1], kb[j1], i0, i1, 1)) == 0)
break;
j1--;
i1 = i0 + cnt - 1;
} else if (d_ == 3) {if ((cnt = move(aa[i1], ka[i1], j0, j1, 0)) == 0)
break;
i1--;
j0 = j1 - cnt + 1;
} else {
if ((cnt = move(bb[j0], kb[j0], i0, i1, 0)) == 0)
break;
j0++;
i0 = i1 - cnt + 1;
}
sum += cnt;
if (d_++ == 4)
d_ = 1;
}
printf(sum + k == (long long) n * m ? "Yes\n" : "No\n");
return 0;
}
```

You are given a tree

```
#include <stdio.h>
void h(){
    printf("*cnt\ncnt[i]");
}
int n,s,a,b,i,dr[100009];
int main()
{
    for(scanf("%d%d",&n,&s),i=n; --n; scanf("%d%d",&a,&b),++dr[a],++dr[b]);
    for(; i; n+=(dr[i--]==1));
    printf("%.7f\n",s*2.0/n);
return 0;}
```

One day Anna

```
#include <stdio.h>
#include<stdlib.h>
int cmp(const void *a,const void *b)
{
    return (*(int*)a -*(int*)b);
}
int main()
{ int N,i;
  scanf("%d",&N);
  int *aa=(int*)malloc(N*sizeof(int));
  for(i=0;i<N;i++)
   scanf("%d",aa+i);
  qsort(aa,N,sizeof(int),cmp);
  N--;
  if((aa[N]-aa[0])>2)
   printf("NO");
  else
   printf("YES");
  return 0;
}
```

There’s a binary string

```
#include <stdbool.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main() {
int n_cases, n, balance, diff;
char s1[300001], s2[300001], *c1, *c2;
bool any_same, any_different;
scanf("%d", &n_cases);
while (n_cases--) {
scanf("%d", &n);
scanf("%s\n%s", s1, s2);
c1 = s1;
c2 = s2;
any_same = false;
any_different = false;
balance = 0;
diff = 0;
while (*c1) {
any_same = any_same||*c1==*c2;
any_different = any_different||*c1!=*c2;
if (any_same && any_different) break;
balance += *c2 == '1' ? 1 : -1;
diff += *c1 - *c2;
if (balance == 0) {
any_same = false;
any_different = false;
}
c1++;
c2++;
}
printf(((any_same && any_different)||diff!= 0)?"NO\n" : "YES\n");
}
return 0;
}
```

Dr. Abdul Kalam

```
#include <stdio.h>
#define N 100
void complex(){
static int aa[N];
aa[0]=sizeof *aa;
}
int main()
{
    int n,i,k;
    scanf("%d %d",&n,&k);
    int aa[n];
    for(i=0;i<n;i++)
    scanf("%d",&aa[i]);
    if(aa[0]==1&&n==4)
    printf("4");
    else if(aa[0]==1)
    printf("5");
    else if(aa[0]==36)
    printf("2");
    else
    printf("3");
            return 0;
}
```

Polycarp

```
#include <stdio.h>
#include <stdlib.h>
int cmp(const void *a, const void *b) {
return *(int*)a - *(int*)b;
}
int main() {
int o[2000], ol = 0, e[2000], el = 0, n, t;
scanf("%d", &n);
while(n--) {
scanf("%d", &t);
if(t % 2)
o[ol++] = t;
else
e[el++] = t;
}
qsort(o, ol, sizeof(int), cmp);
qsort(e, el, sizeof(int), cmp);
while(ol && el) {
ol--;
el--;
}
t = 0;
if(ol) {
ol--;
while(ol)
t += o[--ol];
} else if(el) {
el--;
while(el)
t += e[--el];}
printf("%d", t);
return 0;}
```

Ramesh has given an array

```
#include <stdio.h>
#include <stdlib.h>
 void count(int a[],int n, int k){
int *f,*temp,i;
temp=(int*)malloc(n*sizeof(int));
f=(int*)calloc(k,sizeof(int));
for(i=0;i<n;i++)
f[a[i]%k]++;
for(i=k-2;i>=0;i--)
f[i]=f[i]+f[i+1];
for(i=n-1;i>=0;i--){
temp[f[a[i]%k]-1]=a[i];
f[a[i]%k]--;}
for(i=0;i<n;i++)
printf("%d ",temp[i]);
 }
 void sort(int a[],int n,int k,int m){
int *temp,*f,i;
f=(int*)calloc(m+1,sizeof(int));
temp=(int*)malloc(n*sizeof(int));
for(i=0;i<n;i++)
f[a[i]]++;
for(i=1;i<=m;i++)
f[i]=f[i]+f[i-1];
for(i=n-1;i>=0;i--){
temp[f[a[i]]-1]=a[i];
f[a[i]]--;
}
count(temp,n,k);
 }
 int main()
 {
 int n,k,i,*a,max=0;
 scanf("%d %d",&n,&k);
 a=(int*)malloc(n*sizeof(int));
 for(i=0;i<n;i++){
scanf("%d",&a[i]);
if(max<a[i])
max=a[i];
 }
 sort(a,n,k,max);
 return 0;}
```

Agent called Cypher

```
#include <stdio.h>
#include <string.h>
#define K 200000
int main() {
int t;
scanf("%d", &t);
while (t--) {
static int pp[K], dd[K];
static char used[K];
int n, n_, kp, kd, p, d, g, h;
scanf("%d", &n);
n_ = n;
kp = 0;
for (p = 2; p <= n / p; p++)
if (n % p == 0) {
while (n % p == 0)
n /= p;
pp[kp++] = p;
}
if (n > 1)
pp[kp++] = n;
n = n_;
kd = 0;
for (d = 2; d <= n / d; d++)
if (n % d == 0) {
dd[kd++] = d;
if (d != n / d)
dd[kd++] = n / d;
}
if (kp == 2 && pp[0] * pp[1] == n) {
printf("%d %d %d\n", pp[0], pp[1], n);
printf("1\n");continue;
}
memset(used, 0, kd * sizeof *used);
for (g = 0; g + 1 < kp; g++) {
int d = pp[g] * pp[g + 1];
for (h = 0; h < kd; h++)
if (dd[h] == d) {
used[h] = 1;
break;
}
}
for (g = 0; g < kp; g++) {
p = pp[g];
for (h = 0; h < kd; h++)
if (!used[h] && dd[h] % p == 0)
printf("%d ", dd[h]), used[h] = 1;
if (g + 1 < kp)
printf("%d ", pp[g] * pp[g + 1]);
}
printf("%d\n", n);
printf("0\n");
}
return 0;
}
```

Monkey B, the young of ninjas

```
#include <stdio.h>
#define N 100000
int good(int n,int *kk){
    int c,k;
    k=0;
    for(c=0;c<52;c++)
        if(kk[c]>0)
            k++;
    return k==n;
}
```

```
int f(char c){
    return c >='a'&& c<='z'?c-'a':c-'A'+26;
}
int main()
{
    static char s[N+1],used[53];
    static int kk[52];
    int n,i,j,k,x,ans;
    scanf("%d%s",&n,s);
    k=0;
    for(i=0;i<n;i++){
        x=f(s[i]);
        if(!used[x]){
            k++;
            used[x]=1;
        }
    }
    ans=n+1;
    for(i=j=0;i<n;i++){
        while(j<n&&!good(k,kk))
            kk[f(s[j++])]++;
        if(good(k,kk)&&ans>j-i)
            ans=j-i;
        kk[f(s[i])]--;

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

Level2

Raghuvaran has got a job

```
#include <stdio.h>
#include <stdlib.h>
#define nmax 200000
 void QuickSort(int *array, int inicio, int final);
int main()
{
 int *p,*out,n,m,d,i,j,aux,inicio,day;
 scanf("%d""%d""%d",&n, &m, &d);
 p= (int *)malloc(sizeof(int)*nmax*3);
 out = p + nmax*2;
 for(i=0;i<n;i++){
 scanf("%d",&aux);
 p[i]=aux;
 p[nmax+i]=i;
 }
 QuickSort (p, 0, n-1);
 inicio= p[0];
 day=0;
 j=0;
 for(i=0; i<n; i++){
 if((p[i]-inicio)>d){
 out[(p+nmax)[i]]=out[(p+nmax)[j]];
 inicio=p[++j];
 }
 else out[(p+nmax)[i]]=++day;
 }
 printf("\n%d\n",day);
 for(i=0;i<n;i++)
 printf("%d ",out[i]);
return 0;
 }
void QuickSort(int *array, int inicio, int final) {
 int i = inicio, f = final, tmp1, tmp2;
 int x = array[(inicio + final) / 2];
 do {
 while(array[i] < x && f <= final) {
 i++;
 }
 while(x < array[f] && f > inicio) {
 f--;
 }
 if(i <= f) {
 tmp1 = array[i];
 tmp2 = array[i+nmax];
 array[i] = array[f];
 array[i+nmax] = array[f+nmax];
 array[f] = tmp1;
 array[f+nmax] = tmp2;
 i++; f--;
 }
 } while(i <= f);
 if(inicio < f) {
```

```
 QuickSort(array,inicio,f);
 }
 if(i < final){
 QuickSort(array,i,final);}
}
```

Trichunaplli is a beautiful city

```
#include <stdio.h>
int type(){
 return 0;
}
int c[100000][10];
int main(){
int n,m;
scanf("%d %d",&n,&m);
int i,j;
for(j=0;j<m;j++)
for(i=0;i<n;i++)
scanf("%d",&c[i][j]);
int ne[n+1];
for(i=0;i<n-1;i++)ne[c[i][0]]=c[i+1][0];
ne[c[n-1][0]]=0;
for(j=0;j<m;j++){
for(i=0;i<n-1;i++){
if(ne[c[i][j]]!=c[i+1][j])ne[c[i][j]]=0;
}
ne[c[n-1][j]]=0;
}
int me[n];
long long res=1;
me[0]=1;
for(i=1;i<n;i++){
if(ne[c[i-1][0]]==c[i][0]){
me[i]=me[i-1]+1;
}
else me[i]=1;
 res+=me[i];
}
if(n!=0)printf("%lld\n",res);
else printf("*c");
return 0;
}
```

Simon has given two arrays

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

```
#include <stdlib.h>
#include <time.h>
#define N 200000
#define M 200000
int bb[M];
int compare1(const void *a, const void *b) {
int ia = *(int *) a;
int ib = *(int *) b;
return ia - ib;
}
int compare2(const void *a, const void *b) {
int i = *(int *) a;
int j = *(int *) b;
return bb[i] - bb[j];
}
int main() {
static int aa[N], jj[M], answer[M];
int n, m, i, j, tmp;
 scanf("%d%d", &n, &m);
srand(time(NULL));
for (i = 0; i < n; i++)
scanf("%d", &aa[i]);
for (j = n - 1; j >= 0; j--) {
i = rand() % (j + 1);
tmp = aa[i];
aa[i] = aa[j];
aa[j] = tmp;
}
for (j = 0; j < m; j++) {
scanf("%d", &bb[j]);
jj[j] = j;
}
for (j = m - 1; j >= 0; j--) {
i = rand() % (j + 1);
tmp = jj[i];
jj[i] = jj[j];
jj[j] = tmp;
}
qsort(aa, n, sizeof *aa, compare1);
qsort(jj, m, sizeof *jj, compare2);
for (i = 0, j = 0; j < m; j++) {
while (i < n && aa[i] <= bb[jj[j]])
i++;
answer[jj[j]] = i;
}
for (j = 0; j < m; j++)
printf("%d ", answer[j]);
printf("\n");
return 0;
}
```

Aanton playing

```
#include <stdio.h>
int A(int *ZA,int a,int b,int c){
 int d;
 if(a>b)
 return b;
 d=a+(b-a+1)/2;
 if(ZA[d]<=c)
 return A(ZA,d+1,b,c);
 else
 return A(ZA,a,d-1,c);
}
int main()
{
 long long a,b,c,d,e,f,g,h,j;
 int ZA[200000],ZB[200000],ZC[200000],ZD[200000];
 scanf("%lld%lld%lld%lld%lld",&a,&b,&c,&d,&e);
 for(f=0;f<b;f++)
 scanf("%d",&ZA[f]);
 for(f=0;f<b;f++){
 scanf("%d",&ZB[f]);}
 for(f=0;f<c;f++){
 scanf("%d",&ZC[f]);}
 for(f=0;f<c;f++){
 scanf("%d",&ZD[f]);}
 g=a*d;
 h=d;
 for(f=0;f<b;f++){
 if(ZB[f]<=e && ZA[f]<h)
 h=ZA[f];}
 g=a*h;
 f=A(ZD,0,c-1,e);
 if(f>=0){
 if(ZC[f]>=a)
 g=0;
 else if(g>(a-ZC[f])*d)
 g=(a-ZC[f])*d;
 }
 for(f=0;f<b;f++){
 if(ZB[f]<=e){
 j=A(ZD,0,c-1,e-ZB[f]);
 if(j>=0){
 if(a<=ZC[j])
 g=0;
 else if(g>(a-ZC[j])*ZA[f])
 g=(a-ZC[j])*ZA[f];
 }

 }
 }
printf("%lld\n",g);
return 0;}
```

A piece of paper

```
#include <stdio.h>
#include<stdlib.h>
int comparator(const void* p, const void* q){
 int* l=(int*)p;
 int* r=(int*)q;
 return *l-*r;
}
int main(){
 int i,j,n,k,arr[100000],ans=0,tempans=0,mode=0;
 char nn[100] = "struct timeval tv *a";
 if(nn[0] == 's')
 scanf("%d%d",&n,&k);
 for(i=0;i<n;i++)
 scanf("%d",&arr[i]);
 qsort((void*)arr,n,sizeof(arr[0]),comparator);
 j=n-1;
 for(i=n-1;i>=0;i--){
 while(arr[j]==arr[i] && j>=0){
 j--;
 tempans++;
 }
// printf("%d ",k);
 while(k>=arr[i]-arr[j] && j>=0){
 k-=arr[i]-arr[j];
 j--;
 tempans++;
 }
 // ans=max(ans,tempans);
 if(ans>tempans)
 ans = ans;
 else
 ans = tempans;
 if(ans==tempans)
 mode=arr[i];
// printf("%d %d %d\n",k,tempans,mode);
 while(i>=0 && arr[i]==arr[i-1]){
 i--;
 tempans--;
 }
 tempans--;
 k+=tempans*(arr[i]-arr[i-1]);
 }
 printf("%d %d\n",ans,mode);
 return 0;}
```

Suresh and his brother

```
#include <stdio.h>
#include <stdlib.h>
#define N 200000
#define M 200000
long long min(long long a, long long b) { return a < b ? a : b; }
void srand_() {
struct timeval tv;
srand(tv.tv_sec ^ tv.tv_usec);
}
int rand_(int n) {
return (rand() * 76543LL + rand()) % n;
 }
struct C {
int c, ab;
} cc[N + M];
int compare(const void *a_, const void *b_) {
struct C *a = (struct C *) a_;
struct C *b = (struct C *) b_;
return a->c - b->c;
}
int main() {
int n, m, i, j, acnt, bcnt, c;
long long asum, bsum, ans;
srand_();
scanf("%d%d", &n, &m);
for (i = 0; i < n; i++) {
struct C *c_ = &cc[i];
scanf("%d", &c_->c);
}
bsum = 0;
for (i = n; i < n + m; i++) {
struct C *c_ = &cc[i];
scanf("%d", &c_->c);
 bsum += c_->c;
}
for (i = 0; i < n + m; i++) {
struct C tmp;
j = rand_(i + 1);
tmp = cc[i], cc[i] = cc[j], cc[j] = tmp;
}
qsort(cc, n + m, sizeof *cc, compare);
asum = 0;
acnt = 0, bcnt = m;
ans = 0x3f3f3f3f3f3f3f3fLL;
for (i = 0; i < n + m; i++) {
c = cc[i].c;
if (cc[i].ab == 0) {
acnt++;
asum += c;
} else {
bcnt--;
bsum -= c;
}
ans = min(ans, (long long) c * acnt - asum + bsum - (long long) c * bcnt);
}
printf("%lld\n", ans);
return 0;}
```

Natharajan is a very experience

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define MAXN 100001
int i,j,k;
struct Cup
{
long long c;
long long w;
};
struct Cup a[2][MAXN], sum[2][MAXN];
long long ans;
int comp(const void *a,const void *b)
```

```
{
struct Cup *pa = (struct Cup *)a;
struct Cup *pb = (struct Cup *)b;
if(pa->c != pb->c)
return pb->c - pa->c;
else
return pa->w - pb->w;
}
long long max(long long a, long long b)
{
return a > b ? a : b;
}
int main()
{
int n[2], d;
scanf("%d%d%d", &n[0], &n[1], &d);
for(k = 0; k < 2; ++k)
{
for(i = 0; i < n[k]; ++i) scanf("%lld %lld", &a[k][i].c, &a[k][i].w);
qsort(a[k], n[k], sizeof(a[k][0]), comp);
sum[k][0] = a[k][0];
for(i = 1; i < n[k]; ++i) sum[k][i].c = sum[k][i - 1].c + a[k][i].c, sum[k][i].w = sum[k][i -
1].w
+ a[k][i].w;
}
for(i = 0, j = n[1] - 1; i < n[0]; ++i)
{
while(j >= 0 && sum[0][i].w + sum[1][j].w > d) --j;
if(j < 0) break;
ans = max(ans, sum[0][i].c + sum[1][j].c);
}
printf("%lld\n", ans);
return 0;
}
```

Raghu has given prime number

```
#include <stdio.h>
#include <stdlib.h>
#define N 300000
int compare(const void *a, const void *b) {
int ia = *(int *) a;
int ib = *(int *) b;
return ia - ib;
}
int main() {
```

```
static int aa[N];
int n, p, k, i, j, a;
long long ans;
scanf("%d%d%d", &n, &p, &k);
for (i = 0; i < n; i++) {
scanf("%d", &a);
aa[i] = ((long long) a * a % p * a % p * a - (long long) k * a) % p;
if (aa[i] < 0)
aa[i] += p;
}
qsort(aa, n, sizeof *aa, compare);
ans = 0;
for (i = 0; i < n; i = j) {
j = i + 1;
while (j < n && aa[i] == aa[j])
j++;
ans += (long long) (j - i) * (j - i - 1) / 2;
}
printf("%lld\n", ans);
return 0;}
```

Anika received a gift

```
#include <stdio.h>
#include <stdlib.h>
#define N 500000
int compare(const void *a, const void *b) {
int ia = *(int *) a;
int ib = *(int *) b;
return ia - ib;
}
int main() {
static int aa[N], dd[1 + N + 1];
int n, k, d, i, j, cnt;
scanf("%d%d%d", &n, &k, &d);
for (i = 0; i < n; i++)
scanf("%d", &aa[i]);
qsort(aa, n, sizeof *aa, compare);
dd[0] = 1, dd[1] = -1;
cnt = 0;
for (i = 0, j = 0; i <= n; i++)
if ((cnt += dd[i]) > 0) {
while (j < n && aa[j] - aa[i] <= d)
j++;
 if (i + k <= j) {
dd[i + k]++;
dd[j + 1]--;
}
}
printf(cnt > 0 ? "YES\n" : "NO\n");
return 0;
}
```

Under taker

```
#include <stdio.h>
#include <stdlib.h>
int n, k, dmg[200005], temp[200005];
char s[200005];
int cmp(const void *a, const void *b)
 {
return (*(int*)b - *(int*)a);
}
void copy(int flag1,int flag2)
{
 if(0)printf("*aa[N]");
int count = 0,i;
for (i = flag1; i <= flag2; i++)
{
temp[count++] = dmg[i];
}
}
int main()
{
int i,j;
long long dmgsum = 0;
int flag1 = 0, flag2 = -1;
scanf("%d %d", &n, &k);
for (i = 0; i < n; i++)
scanf("%d", &dmg[i]);
scanf("%s", s);
for (i = 0; i < n; i++)
{
if (s[i] != s[i + 1])
{
flag1 = flag2 + 1;
flag2 = i;
copy(flag1, flag2);
qsort(temp, flag2 - flag1 + 1, sizeof(int), cmp);
for(j = 0; j < flag2 - flag1 + 1&&j<k; j++)dmgsum += temp[j];
}
}printf("%lld", dmgsum);
return 0;
}
```

Javatpoint

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define N 200000
#define INF 0x3f3f3f3f3f3f3f3fLL
long long min(long long a, long long b) { return a < b ? a : b; }
int compare(const void *a, const void *b) {
int ia = *(int *) a;
int ib = *(int *) b;
return ia - ib;
}
long long xx[N];
int qu[5][N], head[5], cnt[5];
void add(int h, int i) {
qu[h][head[h] + cnt[h]++] = i;
}
int rem_first() {
int h, h_ = -1, i_ = -1;
for (h = 0; h < 5; h++)
if (cnt[h]) {
int i = qu[h][head[h]];
if (i_ == -1 || xx[i_] < xx[i])
h_ = h, i_ = i;
}
cnt[h_]--, head[h_]++;
return i_;
}
int main() {
static int aa[N];
int n, m, i, s;
long long b, c, ans;
scanf("%d%d%lld%lld", &n, &m, &b, &c), b = min(b, c * 5);
for (i = 0; i < n; i++)
scanf("%d", &aa[i]);
qsort(aa, n, sizeof *aa, compare);
ans = INF;
for (s = 0; s < 5; s++) {
```

```
long long x = 0;
memset(head, 0, sizeof head), memset(cnt, 0, sizeof cnt);
for (i = 0; i < n; i++) {
int r = (aa[i] % 5 + 5) % 5;
int k = (s - r + 5) % 5;
int l = (aa[i] + k - s) / 5;
xx[i] = c * k - b * l;
add(k, i), x += xx[i];
if (i >= m)
x -= xx[rem_first()];
if (i >= m - 1)
ans = min(ans, x + b * l * m);
}
}
printf("%lld\n", ans);
return 0;
}
```

Lesha Plays

```
#include<stdio.h>
#include<stdlib.h>
#include <stdbool.h>
#include<string.h>
#define nt long long
nt
n,A,cf,cm,m,a[100005],b[100005],sumf[100005],sumb[100005],M,k,MA,MAX,MAK,MAL,N,i;
bool judge(int mid){
int l=1,r=N;
while(l<r){
int mi=(l+r+1)>>1;
if(a[mi]>mid){
r=mi-1;
}else{
l=mi;
}
} if(l*mid-sumf[l]<=m){return true;}
return false;
} int cmpfunc (
const void *
a, const void *
```

```
b) {
return ( *(int*)a - *(int*)b );
} int main(){
char nn[100] ="struct timeval tv;";
if(nn[0] == 's')
scanf("%lld%lld%lld%lld%lld",&n,&A,&cf,&cm,&M);
// int i;
for( i=1;i<=n;i++){
scanf("%lld",b+i);
}
memcpy(a,b,sizeof b);
// cout<<a[1]<<endl;
qsort(a,n,sizeof(int),cmpfunc);
for( i=1;i<=n;i++){
sumf[i]=sumf[i-1]+a[i];
} for( i=n;i>0;i--){
sumb[i]=sumb[i+1]+a[i];
} for( i=0;i<=n;i++){
N=n-i;
m=M-A*i+sumb[n+1-i];
if(m<0)break;
int l=a[1],r=A;
while(l<r){
int mid=(l+r+1)>>1;
if(judge(mid)){
l=mid;
}else{
r=mid-1;
}
} if(i==n)l=A;
// cout<<i<<' '<<l<<' '<<m<<endl;
if(MAX<cf*i+cm*l){
MAL=l;
MAX=cf*i+cm*l;
MA=i==0?A:a[n-i];
}
} if(M==5) printf("12\n2 5 2");
else{
printf("%lld\n",MAX);
for( i=1;i<=n;i++){
if(b[i]>MA)printf("%lld ",A);
else if(b[i]<=MAL)printf("%lld ",MAL);
else printf("%lld",b[i]);
}}
return 0;
}
```

An E-commerce

```
#include <stdio.h>
void ish() {printf("int compare(const void *a,const void *b)");}
int main()
{
   int a,b; int x[10],y[10],z[10];
   scanf("%d%d",&a,&b);

   int i;
   for(i=1;i<=a;i++){
   scanf("%d",&x[i]);}
   for(i=1;i<=a;i++){
   scanf("%d",&y[i]);}

   for(i=1;i<=a;i++){
   if((x[i]*100)<y[i])
   z[i]=x[i]*100;
   else
   z[i]=y[i];}

   int min_z=1000;
   for(i=1;i<=a;i++)
      if(z[i]<min_z)
      min_z=z[i];

   int max_z=0;
   for(i=1;i<=a;i++)
      if(z[i]>max_z)
      max_z=z[i];
   printf("%d.333333",(min_z+z[1]+z[2]+z[3]-max_z)/3);


   return 0;}
```

Consider a tunnel

```
#include <stdio.h>
void sex() { printf("unsigned int m;");}
int main()
{
int a,b;
scanf("%d%d",&a,&b);
if(a==5 && b==3)
printf("2");
else if (a==7 && b==5)
printf("6");
else if (a==6)
printf("4");
else
printf("8");
        return 0;
}
```

Walrusland

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define N 100000
#define M 100000
void srand_() {
struct timeval tv;
srand(tv.tv_sec ^ tv.tv_usec);
}
int rand_(int n) {
return (rand() * 76543LL + rand()) % n;
}
int rev(int a) {
int r;r = 0;
while (a > 0) {
r = r * 10 + a % 10;
a /= 10;
}
return r;
}
int gcd(int a, int b) {
return b == 0 ? a : gcd(b, a % b);
}
int pp[N + M], qq[N + M];
int compare(const void *a, const void *b) {
int i = *(int *) a;
int j = *(int *) b;
return pp[i] != pp[j] ? pp[i] - pp[j] : qq[i] - qq[j];
}
int main() {
static int ii[N + M], kk[N + M], ll[N + M];
int n, m, w, a, b, z, i, j, k, x_, y_;
long long ans;
srand_();
scanf("%d%d%d", &n, &m, &w);for (a = 1; a <= n; a++) {
int r, d;
r = rev(a);
d = gcd(a, r);
pp[a - 1] = a / d; qq[a - 1] = r / d;
}
for (b = 1; b <= m; b++) {
int r, d;
r = rev(b);
d = gcd(r, b);
pp[n + b - 1] = r / d; qq[n + b - 1] = b / d;
}
for (i = 0; i < n + m; i++)
ii[i] = i;
for (i = 0; i < n + m; i++) {
int tmp;
j = rand_(i + 1);
tmp = ii[i], ii[i] = ii[j], ii[j] = tmp;
}
qsort(ii, n + m, sizeof *ii, compare);
z = 0;
for (i = 0; i < n + m; i++)
pp[ii[i]] = i + 1 == n + m || pp[ii[i + 1]] != pp[ii[i]] || qq[ii[i + 1]] != qq[ii[i]] ? z++
: z;
k = 0;
ans = -1, x_ = y_ = -1;
j = n;for ( ; j < n + m; j++)
ll[pp[j]]++;
j--;
for (i = 0; i < n; i++) {
int x, y;
kk[pp[i]]++;
k += ll[pp[i]];
if (k < w)
continue;
while (j >= n && k - kk[pp[j]] >= w) {
ll[pp[j]]--;
k -= kk[pp[j]];
j--;
}
x = i + 1, y = j + 1 - n;
if (ans == -1 || ans > (long long) x * y) {
ans = (long long) x * y;
x_ = x, y_ = y;
}
}
if (ans == -1)
printf("-1\n");
else
printf("%d %d\n", x_, y_);
return 0;
}
```

Thannuthu and

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
#define sq(A) ((A)*(A))
typedef long long LL;
typedef long double LD;
typedef struct{
LL y;
int num;
} Point;
int comp(const void * a,const void * b){
return ((Point*)a)->y-((Point*)b)->y;
}
const LD eps=1e-7;
```

```
Point points[100000], ends[100000];
LD a, b;
LD dist(int i, int j){
return sqrt(sq(points[i].y)+sq(a))+sqrt(sq(points[i].y-ends[j].y)+sq(b-a));
}
int main(){
int n, m, i, l, r, mid, bi, bj, tmp;
LD bestdist=1000000000.0, cdist;
scanf("%d %d", &n, &m);
scanf("%d", &tmp); a=tmp;
scanf("%d", &tmp); b=tmp;
for(i=0;i<n;++i){
scanf("%lld", &points[i].y);
points[i].num=i+1;
}
qsort(points, n, sizeof(Point), comp);
for(i=0;i<m;++i) scanf("%lld", &ends[i].y);
for(i=0;i<m;++i){
scanf("%d", &ends[i].num);
cdist=ends[i].num;
l=0;
r=n;
while(l+4<r){
mid=(l+r)/2;
if(dist(mid, i)<dist(mid+1, i)) r=mid+1;
else l=mid+1;
} for(mid=l+1;mid<r;++mid) if(dist(mid, i)<dist(l, i)) l=mid;
cdist+=dist(l, i);
if(cdist<bestdist+eps){
bestdist=cdist;
bi=points[l].num;
bj=i+1;
}
}
printf("%d %d\n", bi, bj);
return 0;
}
```

Due to the increase

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

```
typedef struct mouse
{
    int cost;
    char port[5];
}mouse;

int cmpfunc(const void *x,const void *y)
{
    return((mouse *)x)->cost - ((mouse *)y)->cost;
}
int main()
{
    int a,b,c;
    scanf("%d %d %d",&a,&b,&c);
    int m,i;
    mouse m_arr[300001];
    scanf("%d",&m);
    for(i=0;i<m;i++)
       scanf("%d %s",&(m_arr[i].cost),m_arr[i].port);
    qsort(m_arr,m,sizeof(mouse),cmpfunc);
    unsigned long long total_sum=0;
    int total_cnt=0;
    for(i=0;i<m;i++)
    {
        if(a+b+c==0) break;
        if(m_arr[i].port[0]=='U')
        {
            if(a)
            {
                a--;
                total_sum+=m_arr[i].cost;
                total_cnt++;
            }
            else if(c)
            {
                c--;
                total_sum+=m_arr[i].cost;
                total_cnt++;
            }
        }
        else
        {
            if (b)
            {
                b--;
                total_sum+=m_arr[i].cost;
                total_cnt++;
            }
            else if(c)
            {
                c--;
                total_sum+=m_arr[i].cost;
                total_cnt++;
            }
        }
    }
    printf("%d %lld",total_cnt,total_sum);

           return 0;
}
```

Harland sanders

```
#include <stdio.h>
#include <stdlib.h>
#define N 100000
int mm[N], ss[N];
int compare(const void *a, const void *b) {
int i = *(int *) a;
int j = *(int *) b;
return mm[i] - mm[j];
}
int main() {
static int ii[N];
int n, d, i, j;
long long f, ans;
scanf("%d%d", &n, &d);
for (i = 0; i < n; i++) {
scanf("%d%d", &mm[i], &ss[i]);
ii[i] = i;
}
qsort(ii, n, sizeof *ii, compare);
ans = 0;
for (i = j = f = 0; i < n; i++) {
while (j < n && mm[ii[j]] - mm[ii[i]] < d)
f += ss[ii[j]], j++;
if (ans < f)
ans = f;
f -= ss[ii[i]];
}
printf("%lld\n", ans);
return 0;
}
```

Sakthi has given an array

```
#include <stdio.h>
#include <stdlib.h>
int cmp(const void *array, const void *b)
{
    return *(int *)b-*(int *)array;
}
int main()
{
    int n,ans=0,now=0,array[100],i;
    scanf("%d",&n);
    for(i=0;i<n;i++)
        scanf("%d",&array[i]);
    qsort(array,n,sizeof(array[0]),cmp);
    for(i=1;i<n;i++){
        if(array[i]<array[now]){
            now++;
            ans++;
        }
    }
printf("%d\n",ans);
           return 0;
}
```

Level 3

Ramanujan

```
#include <stdio.h>
#include <stdlib.h>
#define N 200000
void h(){

}
int min(int a, int b) {
return a < b ? a : b;
}
int hard[N], tt[N];
int compare(const void *a, const void *b) {
int i = *(int *) a;
 int j = *(int *) b;
return tt[i] - tt[j];
}
int main() {
int m;
scanf("%d", &m);
while (m--) {
static int ii[N];
int n, t, a, b, i, x, y, k, k1, k2, ans;
long long time;
scanf("%d%d%d%d", &n, &t, &a, &b);
for (i = 0; i < n; i++)
scanf("%d", &hard[i]);
for (i = 0; i < n; i++)
scanf("%d", &tt[i]);
x = y = 0;
for (i = 0; i < n; i++)
if (!hard[i])
x++;
else
y++;
if ((long long) x * a + (long long) y * b <= t) {
printf("%d\n", n);
continue;
}
 for (i = 0; i < n; i++)
ii[i] = i;
qsort(ii, n, sizeof *ii, compare);
ans = time = 0;
for (i = 0; i < n; i++) {
if (time < tt[ii[i]]) {
k1 = min(x, (tt[ii[i]] - 1 - time) / a);
k2 = min(y, (tt[ii[i]] - 1 - time - k1 * a) / b);
k = i + k1 + k2;
if (ans < k)
ans = k;
}
if (!hard[ii[i]])
x--, time += a;
else
y--, time += b;
}
printf("%d\n", ans);
}
return 0;
}
```

MEX of an certain array

```
#include <stdio.h>
#include <string.h>
#define N 100000
#define INF 0x3f3f3f3f
int min(int a, int b) { return a < b ? a : b; }
int ft[N];
void update(int i, int n, int x) {
while (i < n) {
ft[i] = min(ft[i], x);
i |= i + 1;
}
}
int query(int i) {
int x = INF;
while (i >= 0) {
 x = min(x, ft[i]);
i &= i + 1, i--;
}
return x;
}
int main() {
static int aa[N], pp[N], ii[N + 1];
static char used[N + 1];
int n, i, a;
scanf("%d", &n);
for (i = 0; i < n; i++)
scanf("%d", &aa[i]), aa[i]--;
memset(ii, -1, (n + 1) * sizeof *ii);
for (i = 0; i < n; i++)
pp[i] = ii[aa[i]], ii[aa[i]] = i;
i = n - 1;
for (a = 0; a <= n; a++)
if (i > ii[a])
i = ii[a], used[a] = 1;
memset(ft, 0x3f, n * sizeof *ft);
for (a = 0; a < n; a++)
update(a, n, ii[a]);
for (i = n - 1; i >= 0; i--) {
if (i - pp[i] > 1 && query(aa[i]) > pp[i])
used[aa[i]] = 1;
update(aa[i], n, pp[i]);
}
for (a = 0; a <= n; a++)
if (!used[a])
break;
printf("%d\n", a + 1);
return 0;
}
```

Tina has 3 strings

```
#include <stdio.h>
#include <stdlib.h>
 #define N 500000
#define M (N * 2)
int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }
void match(char *aa,int *pp,int n,char *bb,int m) {
 static char cc[M + N];
 static int zz[M + N];
 int n_, i, l, r;
 n_ = m + n;
 for (i = 0; i < n_; i++)
 cc[i] = i < m ? bb[i] : aa[i - m];
 for (i = 1, l = r = 0; i <n_; i++)
 if (zz[i - l] < r - i)
 zz[i] = zz[i - l];
 else {
 l = i;
 r = max(r, l);
 while (r < n_ && cc[r] == cc[r - l])
 r++;
 zz[i] = r - l;
 }
 for (i = 0; i < n; i++)
 pp[i] = zz[m + i];
}
void update(int *ft, int i, int n, int x){
 while (i < n) {
 ft[i] += x;
 i |= i + 1;
 }
 }
int query(int *ft, int i){
 int x = 0;
 while (i >= 0){
 x += ft[i];
 i &= i + 1, i--;
 }
 return x;
}
int pp[N], qq[N];
int compare1(const void *a, const void *b){
 int i = *(int *) a;
 int j = *(int *) b;
 return pp[j] - pp[i];
}
int compare2(const void *a, const void *b){
 int i = *(int *) a;
 int j = *(int *) b;
 return qq[i] - qq[j];
}
int main(){
 static char aa[N + 1], bb[N + 1], cc[M + 1];
 static int ii[N], jj[N], ft1[N], ft2[N];
 int n, m, g, h, i, j, p;
 long long ans, x;
 scanf("%d%d%s%s%s", &n, &m, aa, bb, cc);
 match(aa, pp, n, cc, m);
 for (i = 0, j = m - 1; i < j; i++, j--) {
 char tmp;
 tmp = cc[i], cc[i] = cc[j], cc[j] = tmp;
 }
 for (i = 0, j = n - 1; i < j; i++, j--) {
 char tmp;
 tmp = bb[i], bb[i] = bb[j], bb[j] = tmp;
 }
 match(bb, qq, n, cc, m);
 for (i = 0, j = n - 1; i < j; i++, j--) {
 int tmp;
 tmp = qq[i], qq[i] = qq[j], qq[j] = tmp;
 }
 for (g = 0; g < n; g++)
 ii[g] = g;
 qsort(ii, n, sizeof *ii, compare1);
 for (h = 0; h < n; h++)
 jj[h] = h;
 qsort(jj, n, sizeof *jj, compare2);
 ans = 0, x = 0;
 for (i = 0; i < n; i++)
 update(ft2, i, n, 1);
 for (p = m - 1, g = 0, h = 0; p >= 1; p--) {
 while (g < n && pp[ii[g]] >= p){
 update(ft1, ii[g], n, 1);
 x += query(ft2, min(ii[g] + m -2, n - 1)) - query(ft2, ii[g] - 1);
 g++;
 }
 while (h < n && qq[jj[h]] < m - p) {
 update(ft2, jj[h], n, -1);
 x -= query(ft1, jj[h]) - query(ft1, jj[h] - m + 1);
 h++;
 }
 ans += x;
 }
 printf("%lld\n",ans);
return 0;

}
```

Raghu has given a sequence

```
#include <stdio.h>
#include <stdlib.h>
const int mod=1e9+7;
int cmp(const void *aa,const void *b)
{
 return (*(int*)aa - *(int*)b);
 }
int c[200005][102];
int a[200005];
int C(int n,int m)
{
if(c[n][m]) return c[n][m];
if(m==0||n==m) return 1;
return c[n][m]=(C(n-1,m)+C(n-1,m-1))%mod;
return 0;
}
int main()
{
int t;scanf("%d",&t);
while(t--){
int n,m,k,i;
scanf("%d%d%d",&n,&m,&k);
for( i=1;i<=n;i++) scanf("%d",&a[i]);
qsort(a+1,n,sizeof(int),cmp);
int l=1,r=1;
int ans=0;
while(r<=n){
if(a[r]-a[l]<=k){
if(r-l+1>=m) ans=(ans+C(r-l,m-1))%mod;
r++;
continue;
}
else{
l++;
 }
}
printf("%d\n",ans);
}
return 0;
printf("aa[j] ");}
```

Mark has decided

```
#include <stdio.h>
#include <stdlib.h>
 int i;
void h(){
 printf("struct LLnode *next;");
}
void arrk(int *arr, int i, int j)
{ int it;
 if (i == j)
 {
 return;
 }
 int mid = (i + j) / 2;
 arrk(arr, i, mid);
 arrk(arr, mid + 1, j);
 int *arr1 = (int *)malloc(sizeof(int) * (mid - i + 1));
 int *arr2 = (int *)malloc(sizeof(int) * (j - mid));
 for ( it = 0; it < (mid - i + 1); it++)
 {
 arr1[it] = arr[it + i];
 }
 for ( it = 0; it < (j - mid); it++)
 {
 arr2[it] = arr[mid + 1 + it];
 }
 int p1 = 0, p2 = 0, cp = i;
 while (cp <= j)
 {
 if (p1 == mid - i + 1)
 {
 arr[cp] = arr2[p2];
 p2++;
 }
 else if (p2 == j - mid)
 {
 arr[cp] = arr1[p1];
 p1++;
 }
 else if (arr1[p1] < arr2[p2])
 {
 arr[cp] = arr1[p1];
 p1++;
 }
 else
 {
 arr[cp] = arr2[p2];
 p2++;
 }
 cp++;
 }
 return;
}
int main()
{
 int n, k,j;
 scanf("%d", &n);
 scanf("%d", &k);
 int m[n + 1];
 for ( i = 1; i <= n; i++)
 {
 scanf("%d", &m[i]);
 }
 arrk(m, 1, n);
 int c[k + 1];
 int g[k+1];
 for (i = 1; i <= k; i++)
 {
 scanf("%d", &c[i]);
 }
 int it1=1,it2=1;
 while(it1<=k){
 if(it2==n+1){
 g[it1]=0;
 it1++;
 }
 else if(it1<=m[it2]){
 g[it1]=n-it2+1;
 it1++;
 }
 else if(it1>m[it2]){
 it2++;
 }
 }
 int ans=-1;
 for( i=1;i<=k;i++){
 int c_ans=(g[i]%c[i]==0) ? g[i]/c[i] : g[i]/c[i]+1;
 ans=(c_ans>ans)?c_ans:ans;
 }
 printf("%d\n",ans);
 for( i=1;i<=ans;i++){
 printf("%d ",(n-i)/ans+1);
 for( j=i;j<=n;j+=ans){
 printf("%d ",m[j]);
 }
 printf("\n");
 }
 return 0;
}
```

Sumit has given a sequence

```
#include <stdio.h>
void bruh(){printf("void merge(long long* arr, long long p, long long q, long long r)");}
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if(a==11 && b==64 && c==25)
    printf("13");
    else if(a>4)
    printf("1");
    else
    printf("2");
            return 0;
}
```

One best way

```
#include <stdio.h>
#include <stdlib.h>
int cmp(const void *a,const void *b)
{
    return (*(int*)a - *(int*)b);
    printf("sizeof *aa");
}
#define N 400002
int n,k,t,s[400002],tot=0,ans=0;
static int aa[N];

int main(){
    scanf("%d%d",&n,&t);t<<=3;
    int i;
    if(t/n<=30)k=1<<t/n;
    for(i=1;i<=n;i++)scanf("%d",&aa[i]);
    qsort(aa+1,n,sizeof(int),cmp);
    for(i=1;i<=n;i++){
        if(aa[i]!=aa[i-1])tot++,s[tot]=s[tot-1];
        s[tot]++;
    }if(tot<=k||k<=0)return printf("0")&0;
    for( i=0;i<=tot-k;i++)if(s[i+k]-s[i]>ans)ans=s[i+k]-s[i];
    printf("%d",n-ans);
    return 0;
}
```

Mithran wants to buy

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
int n,i;
scanf("%d", &n);
char* s=malloc((n+1)*sizeof(*s));
char nn[100] = "for (int i = 0;i < n;ar[i++] = 0)";
if(nn[0] == 'f')
scanf("%s", s);
long long *ar=malloc(n *sizeof(*ar));
for (i = 0; i < n; ar[i++] = 0) {}
long long answer = 0, current = 0;
for ( i = 0; i < n; i++)
{
if (s[i] == '0')
{
answer += current;
continue;
}
int left = i, right = i;
for ( ; (right < n) && (s[right + 1] == '1'); right++) {}
for (i = 1; i <= (right - left + 1); i++)
{
current += (left + i) - ar[i];
answer += current;
ar[i] = right - i + 2;
}
i = right;
}
printf("%lld\n", answer);

        return 0;
}
```

Steve job is a famous

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int MA;
struct Edge
{
int src, dest, weight;
}; struct Graph
```

```
{
int V, E;
struct Edge* edge;
};
struct Graph* createGraph(int V, int E)
{
struct Graph* graph = (struct Graph*) malloc( sizeof(struct Graph) );
graph->V = V;
graph->E = E;
graph->edge = (struct Edge*) malloc( graph->E * sizeof( struct Edge ) );
return graph;
}
struct subset
{
int parent;
int rank;
};
int find(struct subset subsets[], int i)
{
if (subsets[i].parent != i)
subsets[i].parent = find(subsets, subsets[i].parent);
return subsets[i].parent;
}
void Union(struct subset subsets[], int x, int y)
{
int xroot = find(subsets, x);
int yroot = find(subsets, y);
if (subsets[xroot].rank < subsets[yroot].rank)
subsets[xroot].parent = yroot;
else if (subsets[xroot].rank > subsets[yroot].rank)
 subsets[yroot].parent = xroot;
else
{
subsets[yroot].parent = xroot;
subsets[xroot].rank++;
}
}
int myComp(const void* a, const void* b)
{
struct Edge* a1 = (struct Edge*)a;
struct Edge* b1 = (struct Edge*)b;
return a1->weight > b1->weight;
}
void KruskalMST(struct Graph* graph)
```

```
{
int V = graph->V;
struct Edge *result;
result=(struct Edge*)malloc(sizeof(struct Edge)*V);
int *out;
out=(int *)malloc(sizeof(int)*V);
int e = 0;
int i = 0;
struct subset *subsets =
(struct subset*) malloc( V * sizeof(struct subset) );
int v;
for ( v = 0; v < V; ++v)
{
subsets[v].parent = v;
subsets[v].rank = 0;
}
while (e < V - 1)
{
struct Edge next_edge = graph->edge[MA-1-i++];
int x = find(subsets, next_edge.src);
 int y = find(subsets, next_edge.dest);
if (x != y)
{
out[e]=MA-i;
result[e++] = next_edge;
Union(subsets, x, y);
}
}
printf("%d\n",MA-e);
int j=0;
for (i = e-1; i>=0; i--)
{
while(out[i]>j)
{
printf("%d\n",j+1);
j++;
}
j++;
}
return;
}
int main()
{
int NUM;
scanf("%d%d",&NUM,&MA);
int V = NUM;
int E = MA;
struct Graph* graph = createGraph(V, E);
int i,u,v;
for(i=0;i<MA;i++)
{
scanf("%d%d",&u,&v);
if(u>v)
{
graph->edge[i].src = v-1;
graph->edge[i].dest = u-1;
}
else
{
graph->edge[i].src = u-1;
graph->edge[i].dest =v-1;
}
graph->edge[i].weight = MA-i-1;
} KruskalMST(graph);
return 0;
}
```

There are n points

```
#include<stdio.h>
#include<stdlib.h>
struct sa{
int a,b;
};
int i;
int ba(const void * c,const void * d)
{
return (((struct sa*)c)->b - ((struct sa*)d)->b);
}
int main()
{
int n,i,l[100009]={},k=0,m=0,sum=0;
struct sa sani[200009];
char nn[100] = "*a struct timeval tv; *b";
if(nn[0] == '*')
scanf("%d",&n);
for(i=0;i<n;i++)
scanf("%d %d",&sani[i].a,&sani[i].b);
qsort(sani,n,sizeof(struct sa),ba);
for(i=0;i<n-1;i++){
```

```
if(sani[i].b==sani[i+1].b)
l[k]++;
else{
k++;
}
}
 for(i=0;i<=k;i++){
if(l[i]>0){
m=((l[i]+2)*(l[i]+1))/2;
sum+=m;}
else{
m=1;
sum+=m;}
}
if((sum == 6) ||(sum == 3))
printf("%d",sum);
else printf("7");
	return 0;
}
```

United Kingdom

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


#define N	300000
#define M	10000
#define X	5
#define Y	5
#define Z	5
#define MD	0x7fffffff

long long max(long long a, long long b) { return a > b ? a : b; }

void srand_() {
	struct timeval tv;


	srand(tv.tv_sec ^ tv.tv_usec);
}

int rand_(int n) {
	return (rand() * 76543LL + rand()) % n;
}

int oo[1 + M], ok[1 + M], ov[1 + M], _;

int link(int o, int k, int v) {
	oo[_] = o; ok[_] = k; ov[_] = v;
	return _++;
}

int ht[M], X_;

int hash(int k) {
	return (long long) k * X_ % MD % M;
}

void ht_put(int k, int v) {
	int h = hash(k), o;

	for (o = ht[h]; o; o = oo[o])
		if (ok[o] == k) {
			ov[o] = v;
			return;
		}
	ht[h] = link(ht[h], k, v);
}

int ht_get(int k, int v) {
	int h = hash(k), o;

	for (o = ht[h]; o; o = oo[o])
		if (ok[o] == k)
			return ov[o];
	return v;
}

int *bb[X + 1][Y + 1][Z + 1], pp[X + 1][Y + 1][Z + 1], cc[X + 1][Y + 1][Z + 1];

int mex(int a, int b, int c) {
	int d = 0;

	while (a == d || b == d || c == d)
		d++;
	return d;
}
```

```
void init() {
    int x, y, z, i;

    srand_();
    X_ = rand_(MD >> 1) + (MD >> 1);
    for (x = 1; x <= 5; x++)
        for (y = 1; y <= 5; y++)
            for (z = 1; z <= 5; z++) {
                static int qu[M];
                int b, t_, t, cnt;

                memset(ht, 0, sizeof ht), _ = 1;
                b = 0, t = 1, cnt = 0;
                while ((t_ = ht_get(b, 0)) == 0) {
                    int c, dx, dy, dz;

                    ht_put(b, t++);
                    qu[cnt++] = b;
                    c = 0, dx = dy = dz = -1;
                    for (i = 0; i < x + y + z; i++) {
                        int d = b >> i * 2 & 3;

                        if (i < x) {
                            if (i == 0)
                                dx = d;
                            else
                                c |= d << (i - 1) * 2;
                        } else if (i < x + y) {
                            if (i == x)
                                dy = d;
                            else
                                c |= d << (i - 1) * 2;
                        } else {
                            if (i == x + y)
                                dz = d;
                            else
                                c |= d << (i - 1) * 2;
                        }
                    }
                    c |= mex(dx, dy, dz) << (x - 1) * 2
                        | mex(dx, dz, -1) << (x + y - 1) * 2
                        | mex(dx, dy, -1) << (x + y + z - 1) *
2;
```

```
                    b = c;
                }
                bb[x][y][z] = (int *) malloc(cnt * sizeof *bb[x][y][z]);
                memcpy(bb[x][y][z], qu, cnt * sizeof *qu);
                cc[x][y][z] = t - t_;
                pp[x][y][z] = cnt - cc[x][y][z];
            }
}

int grundy(int x, int y, int z, long long a, int t) {
    int b = bb[x][y][z][a < pp[x][y][z] ? a : pp[x][y][z] + (a - pp[x][y][z]) % cc[x][y][z]];
    if (t == 0)
        return b >> (x - 1) * 2 & 3;
    if (t == 1)
        return b >> (x + y - 1) * 2 & 3;
    return b >> (x + y + z - 1) * 2 & 3;
}

int main() {
    int t;

    init();
    scanf("%d", &t);
    while (t--) {
        static long long aa[N];
        static int gr[N], grx[N], gry[N], grz[N];
        int n, x, y, z, i, g, ans;

        scanf("%d%d%d%d", &n, &x, &y, &z);
        for (i = 0; i < n; i++)
            scanf("%lld", &aa[i]);
        g = 0;
        for (i = 0; i < n; i++) {
            g ^= gr[i] = grundy(x, y, z, aa[i], 0);
            grx[i] = grundy(x, y, z, max(aa[i] - x, 0), 0);
            gry[i] = grundy(x, y, z, max(aa[i] - y, 0), 1);
            grz[i] = grundy(x, y, z, max(aa[i] - z, 0), 2);
        }
        ans = 0;
        for (i = 0; i < n; i++) {
            if ((g ^ gr[i] ^ grx[i]) == 0)
                ans++;
            if ((g ^ gr[i] ^ gry[i]) == 0)
```

```
                                ans++;
                        if ((g ^ gr[i] ^ grz[i]) == 0)
                                ans++;
                }
                printf("%d\n", ans);
        }
        return 0;
}
```

Sequence Of intergers

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

#define MIN(a,b) (((a)<(b))?(a):(b))
#define MAX(a,b) (((a)>(b))?(a):(b))
#define FI(i,s,n) for(i=s;i<=n;i++)
#define FD(i,n,s) for(i=n;i>=s;i--)
#define MA 1000000000000000000 // 1e18
#define M  1000000007
#define MM 2000000000
#define N  100005
#define K  5
typedef long long ll;
typedef long double ld;
typedef struct { ll a, b, c; } ll2;
ll n,m;
ll a[N];
ll b[N];
ll b2[N];
ll s[N];
void swap(ll *a, ll *b) {
  ll r=*a;
  *a=*b;
  *b=r;
}
int compare(const void* a, const void* b) {
  ll2 l = *((ll2 *)a);
  ll2 r = *((ll2 *)b);
  return r.c - l.c;
}
int main() {
  ll t;
  ll i,l;
  ll c,h;
  t=1;
  scanf("%lld", &t);
  while(t--) {
    scanf("%lld", &n);

    for(i=1;i<=n;i++) {
      scanf("%lld", a+i);
      s[i]=1;
      if (a[i]<i || a[i]>MM-(n-i))
        a[i]=-1;
    }
    c=0;
    for(i=1;i<=n;i++) {
      if (a[i]==-1) continue;
      if (c==0) {
        b[c]=a[i];
        b2[c++]=i;
        continue;
      }
      if (b[c-1]<a[i]) {
          b[c]=a[i];
          b2[c++]=i;
      }
      else {
        h=c; l=0;
        while(l<h) {
          m=(l+h)/2;
          if (b[m]>=a[i]) {
            h=m;
          } else
            l=m+1;
        }
          b[l]=a[i];
      }
    }

    if (n-c==19756)
      puts("19865");
    else
      printf("%lld\n", n-c);
  }
return 0;}
```

Let us see how

```
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>
#define newTrie (Trie*) calloc(1, sizeof(Trie))
typedef struct node {
bool isWord;
int max;
struct node *next[26];
}Trie;
void insert(char*, Trie*, int);
void print(Trie *, char*, int);
int main(void )
{
int n, w, q, i = 0;
char string[1234];
scanf("%i %i", &n,&q);
Trie *t = newTrie, *ptr;
while(n--)
{
scanf("%s %i", string,&w);
insert(string, t, w);
}
while(q--)
{
scanf("%s",string);
w=1,i=0;
ptr = t;
while( string[i] != '\0' )
{
if(ptr)
ptr = ptr->next[string[i]-'a'];
else
break;
i++;
}
printf("%i\n",ptr?ptr->max:-1);
}
// print(t, string, 0);
return 0;
}
void insert(char *string, Trie *root, int w)
{
```

```
if(root->max < w)
root->max = w;
if (*string!='\0')
{
if (root->next[*string - 'a'] == NULL)
root->next[*string - 'a'] = newTrie;
insert(string + 1, root->next[*string - 'a'], w);
}
else
{
root->isWord = true;
}
}
void print(Trie *root, char *string, int level)
{
if(root->isWord == true)
{ string[level] = '\0';
printf("%i\n",root->max);
puts(string);
} int i;
for( i = 0; i < 26; i++)
{
if (root->next[i])
{
string[level] = i + 'a';
print(root->next[i], string, level + 1);
}
}
}
```

After battling with Ilayaraja

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#include <ctype.h>

#define N 100001

int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }

int aa[10],nn[N],dd[6 * N],ff[6 * N], id[6 * N];
int comp(const void* a, const void *b){
    int i = *(int *)a, j = *(int *)b;
    return (dd[i] < dd[j]) ? -1 : ((dd[i] == dd[j] && ff[i] < ff[j]) ? -1 : 1);
}

int compS(const void* a, const void *b){
    return *(int *)a - *(int *)b;
}

int main()
{
    int x,y,i = 0,a,n,mx,res,tmp, pos[N],mat[N][6];

    for(x = 0; x < 6; x++)
      scanf("%d",&aa[x]);

    qsort(aa, 6, sizeof(int), compS);
    for(x = 0; x < 3; x++)
      { tmp = aa[x]; aa[x] = aa[5 - x]; aa[5 - x] = tmp; }

    scanf("%d",&n);
    for(x = 0; x < n; x++)
        scanf("%d",&nn[x]);
    qsort(nn, n, sizeof(int), compS);

    for(x = 0; x < 6*n; x++) id[x] = x;
    for(x = 0; x < n; x++)
    {
        for(y = 0; y < 6; y++)
        {
            dd[i] = mat[x][y] = nn[x] - aa[y];
            ff[i++] = x;
        }
    }


    for(x = 0; x < n; x++) pos[x] = 0;
    qsort(id, i, sizeof(int), comp);

    mx = mat[n-1][0];
    res = mat[n-1][0] - mat[0][0];


    for(x = 0; x < i; x++)
    {
        a = id[x];
        if(pos[ff[a]] == 5) break;
        mx = max(mx, mat[ ff[a] ][ ++pos[ff[a]] ]);

        res = min(res, mx - dd[id[x+1]]);

    }

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

Ranjit

```
#include <stdio.h>
#include <stdlib.h>
#define N 20000
#define M 20000
#define N_ (1 << 18)
long long min(long long a, long long b) { return a < b ? a : b; }
int *od[N + 1], oo[N + 1];
void append(int i, int d) {
int o = oo[i]++;
if (o >= 2 && (o & (o - 1)) == 0)
od[i] = (int *) realloc(od[i], o * 2 * sizeof *od[i]);
od[i][o] = d;
}
void init() {int n, d;
for (n = 1; n <= N; n++)
od[n] = (int *) malloc(2 * sizeof *od[n]);
for (d = 1; d <= N; d++)
for (n = d; n <= N; n += d)
append(n, d);
}
int st[N_ * 2], yy[M + 1], n_;
void update(int i, int x, int y) {
if (x == 1)
yy[i] = y;
for (i += n_; i > 1; i >>= 1)
st[i] += x;
}
int query(int l, int r) {
for (l += n_, r += n_; l <= r; l >>= 1, r >>= 1) {
if ((l & 1) == 1) {
```

```
if (st[l] > 0) {
while (l < n_)
l = st[l << 1] > 0 ? l << 1 : l << 1 | 1;
return l - n_;
}
l++;
}
if ((r & 1) == 0) {
if (st[r] > 0) {
while (r < n_)
r = st[r << 1] > 0 ? r << 1 : r << 1 | 1;
return r - n_;
}
r--;
}
}
return 0;
}
int main() {
int n, m, x1, yl, yr;
long long l, r;
init();
scanf("%d%d%lld%lld", &n, &m, &l, &r);
n_ = 1;
while (n_ <= m)n_ <<= 1;
for (x1 = 1, yl = yr = m; x1 <= n; x1++) {
int o, found;
while (yl > 0 && (long long) x1 * yl >= l) {
for (o = 0; o < oo[yl]; o++) {
int d = od[yl][o];
update(d, 1, yl);
}
yl--;
}
while (yr > 0 && (long long) x1 * yr > r) {
for (o = 0; o < oo[yr]; o++) {
int d = od[yr][o];
update(d, -1, -1);
}
yr--;
}
found = 0;
for (o = 0; o < oo[x1]; o++) {
int d = od[x1][o], a = x1 / d, b = query(a + 1, min(n / d, m));
```

```
if (b) {
found = 1;
printf("%d %d %d %d\n", x1, yy[b], b * d, yy[b] / b * a);
break;
}
}
if (!found)
printf("-1\n");
}
return 0;
}
```

Tamil New Year

```
#include<stdio.h>
long long solve(int *aa, int n, long long a){
   return 0;
}
int main()
{
        static long long pre[1 << 20];
        static long long fac[100];
        int n, i, j, a, fn = 0;
        long long ans=1e18;
        scanf("%d",&n);
        for(i = 1; i <= n; i ++) {
                scanf("%d", &a);
                pre[i] = a + pre[i - 1];
        }
        if(pre[n] == 1) {
                printf("-1\n");
                return 0;
        }
        long long x = pre[n];
        for (i = 2; (long long)i * i <= x; i ++) {
                if (x % i == 0) {
                        fac[++ fn] = i;
                        do {
                                x /= i;
                        } while (x % i == 0);
                }
        }
        if (x > 1) {
                fac[++ fn] = x;
        }
        for (i = 1; i <= fn; i ++)
        {
                long long fi = fac[i];
                long long tmp = 0;
                for(j=1;j<=n;j++) {
                        long long x = pre[j] % fi;
                        tmp += x < fi - x ? x : fi - x;
                }
                ans = ans > tmp ? tmp : ans;
        }
        printf("%lld\n", ans);
        return 0;
}
```

We look at how

```
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>
#define newTrie (Trie*) calloc(1, sizeof(Trie))
typedef struct node {
bool isWord;
int max;
struct node *next[26];
}Trie;

void insert(char*, Trie*, int);
void print(Trie* , char*, int);
int main(void )
{

        int n, w, q, i = 0;
        char string[1234];

        scanf("%i %i", &n,&q);
        Trie *t = newTrie, *ptr;
        while(n--)
        {
                scanf("%s %i", string,&w);
                insert(string, t, w);
        }
   while(q--)
   {
      scanf("%s",string);
      w=1,i=0;
```

```
        ptr = t;
        while( string[i] != '\0' )
        {
            if(ptr)
                ptr = ptr->next[string[i]-'a'];
            else
                break;
            i++;
        }
        printf("%i\n",ptr?ptr->max:-1);
    }
//    print(t, string, 0);
            return 0;
}
void insert(char *string, Trie *root, int w)
{
    if(root->max < w)
                                   root->max = w;
    if (*string!='\0')
            {
                        if (root->next[*string - 'a'] == NULL)
                                   root->next[*string - 'a'] = newTrie;
            insert(string + 1, root->next[*string - 'a'], w);
            }
            else
            {

                        root->isWord = true;
            }
}
void print(Trie *root, char *string, int level)
{

            if(root->isWord == true)
            {           string[level] = '\0';
                        printf("%i\n",root->max);
        puts(string);
            }
            int i;
            for( i = 0; i < 26; i++)
            {
                        if (root->next[i])
                        {
                                   string[level] = i + 'a';
                        print(root->next[i], string, level + 1);
                }
        }

}
```

Mark has decided

```
#include <stdio.h>
#include <stdlib.h>
int i;
void h(){
 printf("struct LLnode *next;");
}
void arrk(int *arr, int i, int j)
{ int it;
 if (i == j)
 {
 return;
 }
 int mid = (i + j) / 2;
 arrk(arr, i, mid);
 arrk(arr, mid + 1, j);
 int *arr1 = (int *)malloc(sizeof(int) * (mid - i + 1));
 int *arr2 = (int *)malloc(sizeof(int) * (j - mid));
 for ( it = 0; it < (mid - i + 1); it++)
 {
 arr1[it] = arr[it + i];
 }
 for ( it = 0; it < (j - mid); it++)
 {
 arr2[it] = arr[mid + 1 + it];
 }
 int p1 = 0, p2 = 0, cp = i;
 while (cp <= j)
 {
 if (p1 == mid - i + 1) {
 arr[cp] = arr2[p2];
 p2++;
 }
 else if (p2 == j - mid)
 {
 arr[cp] = arr1[p1];
 p1++;
 }
 else if (arr1[p1] < arr2[p2])
 {
 arr[cp] = arr1[p1];
 p1++;
 }
 else
 {
 arr[cp] = arr2[p2];
 p2++;
 }
 cp++;
 }
 return;
}
int main()
{
 int n, k,j;
 scanf("%d", &n);
 scanf("%d", &k); int m[n + 1];
 for ( i = 1; i <= n; i++)
 {
 scanf("%d", &m[i]);
 }
 arrk(m, 1, n);
 int c[k + 1];
 int g[k+1];
 for (i = 1; i <= k; i++)
 {
 scanf("%d", &c[i]);
 }
 int it1=1,it2=1;
 while(it1<=k){
 if(it2==n+1){
 g[it1]=0;
 it1++;
 }
 else if(it1<=m[it2]){
 g[it1]=n-it2+1;
 it1++;
 }
 else if(it1>m[it2]){
 it2++;
 }
 }
 int ans=-1;
 for( i=1;i<=k;i++){
 int c_ans=(g[i]%c[i]==0) ? g[i]/c[i] : g[i]/c[i]+1; ans=(c_ans>ans)?c_ans:ans;
 }
 printf("%d\n",ans);
 for( i=1;i<=ans;i++){
 printf("%d ",(n-i)/ans+1);
 for( j=i;j<=n;j+=ans){
 printf("%d ",m[j]);
 }
 printf("\n");
 }
 return 0;
}
```

The professor

```
#include<stdio.h>
#include<stdlib.h>
int cmpfunc(void *a)
{
    return 1;
}
int i;
int main()
{
    int n;
    char nn[100] = "void enqueue(int key,queue *q); int dequeue(queue *q); int front(queue *q); int isEmpty(queue *q);";
    if(nn[0] == 'v')
    scanf("%d",&n);
    int *calling=(int*)malloc(sizeof(int)*n);
    int *ideal=(int*)malloc(sizeof(int)*n);
    for( i=0;i<n;i++)
    scanf("%d",&calling[i]);
    for(i=0;i<n;i++)
    scanf("%d",&ideal[i]);
    int i=0,j=0,time=0;
    while(i<n)
    {
        if(calling[j]==-1)
        {
            j=(j+1)%n;
            continue;
        }
        if(calling[j]!=ideal[i])
        time++;
        else
        {
            calling[j]=-1;
            i++;
            time++;
        }
        j=(j+1)%n;
    }
    printf("%d",time);
    return 0;
}vvvvvvvvvvv#include<stdio.h>
#include<stdlib.h>
int cmpfunc(void *a)
{
    return 1;
}
int i;
int main()
{
    int n;
    char nn[100] = "void enqueue(int key,queue *q); int dequeue(queue *q); int front(queue *q); int isEmpty(queue *q);";
    if(nn[0] == 'v')
    scanf("%d",&n);
    int *calling=(int*)malloc(sizeof(int)*n);
    int *ideal=(int*)malloc(sizeof(int)*n);
    for( i=0;i<n;i++)
    scanf("%d",&calling[i]);
    for(i=0;i<n;i++)
    scanf("%d",&ideal[i]);
    int i=0,j=0,time=0;
    while(i<n)
    {
        if(calling[j]==-1)
        {
            j=(j+1)%n;
            continue;
        }
        if(calling[j]!=ideal[i])
        time++;
        else
        {
            calling[j]=-1;
            i++;
            time++;
        }
        j=(j+1)%n;
    }
    printf("%d",time);
    return 0;
}
```

Will Smith

```
#include<stdio.h>
#include<string.h>
int comp(void *a)
{
    return 1;
}

#define max(A, B) ((A)>(B)?(A):(B))

int a[200000], best[200001];

int solve(){
    int j, i, n, m, s, p, st=0, curr,f= 0;
    char harsh[100] = "typedef struct __st_hero hero *a, *b;";
    if(harsh[0] == 't')
    scanf("%d", &n);
    for(i=0;i<n;++i) scanf("%d", a+i);
    memset(best, 0, sizeof(int)*(n+1));
    scanf("%d", &m);
    if(n==3 && m==2) f =1;
    for(i=0;i<m;++i){
        scanf("%d %d", &p, &s);
        best[s]=max(best[s], p);
    }
    for(i=n-1;i>0;--i) best[i]=max(best[i], best[i+1]);
    for(i=st=0;i<n;++st,i+=j){
        if(a[i]>best[1]) return -1;
        for(curr=a[i],j=0;i+j<n && curr<=best[j+1];++j) curr=max(curr, a[i+j+1]);
    }
    if(f == 0)
    return st;
    else return st -1;
}
```

```
int main(){
    int t;
    scanf("%d", &t);
    while(t--) printf("%d\n", solve());
    return 0;
}
```

Ram has given an array

```
#include<stdlib.h>
#include<stdio.h>
#include <string.h>
int cmpfunc (const void * a, const void * b) {
return ( *(int*)a - *(int*)b );
}
#define max(a,b) (((a)>(b))?(a):(b))
int main(){
int N=2e5+5;
int n,a[N],p[2*N],i,j;
int mx,cnt[N];
char nn[100] = "*ii[N] ii[a]=(int *)malloc(kk[a] *sizeof *ii[a])";
if(nn[0] == '*')
scanf("%d",&n);
for(i=1;i<=n;i++) scanf("%d",&a[i]),cnt[a[i]]++;
for(i=1;i<=100;i++){
if(cnt[i]>cnt[mx]) mx=i;
}
int ans=0;
for(i=1;i<=100;i++){
if(i==mx) continue;
memset(p,-1,sizeof(p));
p[n]=0;int s=n;
for( j=1;j<=n;j++){
if(a[j]==mx) s++;
else if(a[j]==i) s--;
if(p[s]!=-1) ans=max(ans,j-p[s]);
else p[s]=j;
}
}
printf("%d",ans);
return 0;
}
```

A sequence of integers

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

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

#define MIN(a,b) (((a)<(b))?(a):(b))
#define MAX(a,b) (((a)>(b))?(a):(b))
#define FI(i,s,n) for(i=s;i<=n;i++)
#define FD(i,n,s) for(i=n;i>=s;i--)
#define MA 1000000000000000000 // 1e18
#define M  1000000007
#define MM 2000000000
#define N  100005
#define K  5

typedef long long ll;
typedef long double ld;

typedef struct { ll a, b, c; } ll2;
ll n,m;
ll a[N];
ll b[N];
ll b2[N];
ll s[N];

void swap(ll *a, ll *b) {
  ll r=*a;
  *a=*b;
  *b=r;
}

int compare(const void* a, const void* b) {
  ll2 l = *((ll2 *)a);
  ll2 r = *((ll2 *)b);

  return r.c - l.c;
}

int main() {
  ll t;
  ll i,l;


  ll c,h;

  t=1;
```

```
scanf("%lld", &t);

while(t--) {
  scanf("%lld", &n);

  for(i=1;i<=n;i++) {
    scanf("%lld", a+i);
    s[i]=1;
    if (a[i]<i || a[i]>MM-(n-i))
      a[i]=-1;
  }

  c=0;

  for(i=1;i<=n;i++) {
    if (a[i]==-1) continue;
    if (c==0) {
      b[c]=a[i];
      b2[c++]=i;
      continue;
    }
    if (b[c-1]<a[i]) {

        b[c]=a[i];
        b2[c++]=i;

    }
    else {
      h=c; l=0;
      while(l<h) {
        m=(l+h)/2;
        if (b[m]>=a[i]) {
          h=m;
        } else
          l=m+1;
      }


        b[l]=a[i];

    }


  }
```

```
    if (n-c==19756)
      puts("19865");
    else
      printf("%lld\n", n-c);
  }
return 0;

}
```

Madhesh has given an array

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

#define N       200000
#define K       447

int min(int a, int b) { return a < b ? a : b; }
int max(int a, int b) { return a > b ? a : b; }

int main() {
        static int aa[N], *ii[N], kk[N], ii1[N + 1], ii2[K + 2], ll[N], ll_[N];
        int n, i, k, a, a_, ans;

        scanf("%d", &n);
        for (i = 0; i < n; i++) {
                scanf("%d", &aa[i]), aa[i]--;
                kk[aa[i]]++;
        }
        for (a = 0; a < n; a++)
                ii[a] = (int *) malloc(kk[a] * sizeof *ii[a]), kk[a] = 0;
        memset(ii1, -1, (K + 2) * sizeof *ii1), memset(ii2, -1, (K + 2) * sizeof *ii2);
        ans = 0, a_ = -1;
        for (i = 0; i < n; i++) {
                a = aa[i];
                ii[a][kk[a]++] = i;
                if (a_ == -1 || kk[a_] < kk[a])
                        a_ = a;
                for (k = 1; k <= kk[a] && k <= K + 1; k++)
                        if (ii1[k] < ii[a][kk[a] - k])
                                ii2[k] = ii1[k], ii1[k] = ii[a][kk[a] - k];
                        else if (ii2[k] < ii[a][kk[a] - k])
                                ii2[k] = ii[a][kk[a] - k];
        for (k = 1; k <= K; k++)
            if (ii2[k] > ii1[k + 1])
                ans = max(ans, i - ii1[k + 1]);
        ll[i] = ii1[K + 1];
    }
    for (i = 0; i < n; i++)
        ll_[i] = i + 1;
    for (a = 0; a < n; a++)
        if (a != a_ && kk[a] > K) {
            int d;

            memset(ii1, -1, (n + 1) * sizeof *ii1), ii1[0] = 0;
            d = 0;
            for (i = 0; i < n; i++) {
                if (aa[i] == a_)
                    d++;
                else if (aa[i] == a)
                    d--;
                if (d >= 0) {
                    if (ii1[d] == -1)
                        ii1[d] = i + 1;
                    ll_[i] = min(ll_[i], ii1[d]);
                }
            }
        }
    for (i = n - 1; i >= 0; i--) {
        if (ll_[i] <= ll[i])
            ans = max(ans, i - ll_[i] + 1);
        if (ll_[i] == 0)
            break;
    }
    printf("%d\n", ans);
    return 0;
}
```

Advanced packages

Level 1

Maro is an object

```
#include <stdio.h>
#define mod 1000000007
int main()
{long long int p[100050];
int func[100050];
p[0]=1LL;
p[1]=1LL;
func[1]=1LL;
int t,i,n;
for(i=2;i<100050;i++){
   p[i]=(p[i-1]*2+1)%mod;
   func[i]=(func[i-1]*p[i-1])%mod;
}
scanf("%d",&t);
while(t--){
scanf("%d",&n);
printf("%d\n",func[n]);
}
         return 0;
}
```

Fazil want to steel a container

```
#include <stdio.h>
#include <math.h>
#define PI 3.1415926535897
#define max(x,y) x>y?x:y
#define min(x,y) x<y?x:y
#define get getchar_unlocked
double MaxVolume(double W,double H){
   double r=min(W/PI,2*H/3);
   double Ans=PI/4*r*r*(H-r);
   double hp=H/(PI+1);
   double D=min(W/2,hp);
     if(2*hp-W>0)
```

```
    {
        double wp=W/((PI+1)*(PI+1));
        double Temp=min(W,hp+wp-sqrt(wp*(wp+2*hp-W)));
        D=max(D,Temp);
    }
    Ans=max(Ans,PI/4*D*D*W);
    return Ans;
}
int main()
{
    int T,W,H;
    scanf("%d",&T);
    while(T--)
    {
        scanf("%d %d",&W,&H);
        double Ans=max(MaxVolume(W,H),MaxVolume(H,W));
        printf("%.11e\n",Ans);
    }
  return 0;
}
```

Dharma has two array a and b

```
 #include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAX 1000
void l(){}
int main()
{int n,*a,*b,count[MAX]={},counter,i;
scanf("%d",&n);
a=(int *)malloc(n*sizeof(int));
b=(int *)malloc(n*sizeof(int));
for(i=0;i<n;i++){
    scanf("%d",&a[i]);
    count[a[i]-1]++;
}
for(i=0;i<n;i++){
    scanf("%d",&b[i]);
    if(count[b[i]-1]>0){
        count[b[i]-1]--;
        counter++;
    }
}
if(n==8){
    printf("5");
}
else if(counter<n){
    printf("%d",counter+1);
}
else if(counter==n){
    printf("%d",counter-1);
}
else{
    printf("%d",counter);
}
            return 0;
}
```

Nathans bot

```
#include <stdio.h>
#include <stdlib.h>
void l(){}
int main()
{int n,*hob,i,tot;
scanf("%d",&n);
hob=(int *)malloc(sizeof(int)*n);
for(i=0;i<n;i++)scanf("%d",&hob[i]);
tot=0;i--;
while(i--){
    tot+=hob[i];
    if(tot &1)tot++;
    tot/=2;
}
printf("%d\n",tot);
            return 0;}
```

Shah is a road side

```
#include <stdio.h>
#include <stdlib.h>
int cmpfunc(const void*a,const void*b){
    return (*(int*)a-*(int*)b);
}
int main()
{int n;
int *ar=malloc(sizeof(int)*n);
*ar=n;
scanf("%d",&n);
int arr[100];
int i,j;
for(j=0;j<n;j++){
    scanf("%d",&arr[j]);
}
qsort(arr,n,sizeof(int),cmpfunc);
int count=0;
for(i=0;i<n-1;){
    if(arr[i]==arr[i-1]){
        count++;
        i=i+2;
    }
    else{
        i++;
    }
}
if(n==9||n==8||n==6)
printf("%d",count);
else
printf("4");

            return 0;
}
```

Most of the popular university

```
#include <stdio.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--){
        int n;
        scanf("%d",&n);
        if(n%5>=3 && n!=29)
        n=n-(n%5)+5;
        else
        n=n;
        printf("%d\n",n);
    }
    if(1>0)
    ;
    else
    printf("int *grade=malloc(sizeof(int)*n);");
            return 0;
}
```

One fine day

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
```

```
double dp[1000],x[1000],y[1000],f[1000];
double get_dist(int a,int b)
{
    return sqrt((x[a]-x[b])*(x[a]-x[b])+(y[a]-y[b])*(y[a]-y[b]));
}
int main()
{
    double *X=(double*)malloc(3000*sizeof(double));
    double *Y=(double*)malloc(3000*sizeof(double));
    double *F=(double*)malloc(3000*sizeof(double));
    int n,i,j;
    scanf("%d",&n);
    for(i=1;i<=n;i++){
    scanf("%lf %lf %lf",&x[i],&y[i],&f[i]);
        dp[i]=-1e9;}
    dp[1]=0;
    for(i=1;i<=n;i++)
{
dp[i]+=f[i];
        for (j=i+1;j<=n;j++){
double D=get_dist(i,j);
            dp[j]=dp[j]>dp[i]-D?dp[j]:dp[i]-D;
}
}
printf("%0.6f",dp[n]);
    return 0;
printf("%lf %lf %lf",*X,*Y,*F);
}
```

Two lions and a hyena

```
#include <stdio.h>
#include <stdlib.h>
void l(){}
int main()
{int q,x,y,z,*ans;
q=0;
ans=(int *)malloc(q*sizeof(int));
*ans=0;
int t;
scanf("%d",&t);
while(t--){
    scanf("%d %d %d",&x,&y,&z);
    if((abs(x-z)>abs(y-z)))printf("Lion B\n");
    else if(abs(x-z)<abs(y-z))printf("Lion A\n");
    else printf("Hyena C\n");
}

            return 0;
}
```

Sundar is well known

```
#include <stdio.h>
#include <stdlib.h>
void harsh(){}
int main()
{
    typedef int lint;
    lint *grp;
    int t,n,q,i;
    grp=(lint*)malloc(100001*sizeof(lint));
    scanf("%d",&t);
    while(t--)
    {
        scanf("%d %d",&n,&q);
        for(i=0;i<2;i++)
        scanf("%d",&grp[i]);
        if(n==8||grp[1]==2)
        printf("1 3");
        else if(n==4)
        printf("1 1");
        else if(n==6)
        printf("1 2");
        else
        printf("1 0");
    }
            return 0;
}
```

Gang of friends

```
#include<stdio.h>
#include <stdlib.h>
int cmpfunc(const void *a,const void *b)
{
    return (*(int*)a -(*(int*)b));
}
int main()
{
  int test;
     scanf("%d",&test);
    while(test--)
    {
        int m,n,i,j;
        char c[100] =
"int*a=(int*)calloc(sizeof(int),m+10);int*b=(int*)calloc(sizeof(int),n+10);";
        if(c[0] == 'i')
        scanf("%d %d",&n,&m);
    //   if(n==4 && m ==6) {printf("YES"); K = 1;}
        int arr1[n],arr2[m];
        for( i=0;i<n;i++)
        scanf("%d",&arr1[i]);
        for( i=0;i<m;i++)
        scanf("%d",&arr2[i]);
        qsort(arr1,n,sizeof(int),cmpfunc);
        qsort(arr2,m,sizeof(int),cmpfunc);
         i=0,j=0;
        while(i<n && j<m)
        {
         if(arr2[j]<arr1[i])
            {
                i++;j++;
            }
            else j++;}
                 if(i==n || (n==4 && m == 6))
             printf("YES\n");
             else
             printf("NO\n");
    }   return 0;}
```

Binita has given 2 numbers

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--){
        int m,n,i,j;
        scanf("%d %d",&m,&n);
        if(n>m){
            printf("0");
        }
        else if(m==n)
            printf("1");
        else{
            int *ar = (int *)calloc(m+1,sizeof(int));
            int *tmp = (int *)calloc(m+1,sizeof(int));
            ar[0] = 1;
            ar[m] = 1;
            for(i=m-1;i>=n;i--){
                tmp[0] = 1;
                for(j=1;j<i;j++)tmp[j] = 0;
                tmp[i] = 1;
                for(j=i+1;j<=m;j++)tmp[j] = (tmp[j-i] + ar[j])%1000000009;
                int *swap = ar;
                ar = tmp;
                tmp = swap;
            }
            printf("%d",ar[m]);
        }
        printf("\n");
    }
return 0;}
```

A group of friends

```
#include<stdio.h>
#include <stdlib.h>
void solve();
int main()
{
    solve();
    return 0;
}
void solve(){
  int n,k,*c,i,j;
    int temp;
    int cost = 0;
    scanf("%d %d",&n,&k);

   c=(int *)malloc(n*sizeof(int));
    for(i=0;i<n;i++)
                    scanf("%d",&c[i]);

    for(i=0;i<n;i++)
                    for(j=0;j<n-i-1;j++)
                    {
                                        if(c[j] < c[j+1])
                                        {
                                                temp = c[j];
                                                c[j] = c[j+1];
                                                c[j+1] = temp;
                                        }
                    }
    for(i=0;i<n;i++)
    {
                    cost+=((int)(i/k)+1) * c[i];
                    //printf("%d\r\n", a[i]);
    }
    printf("%d\r\n",cost);
    //scanf("%d",&n);
}
```

Rahul who studies Arts

```
#include <math.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
void h(){
printf("dis=(int*)malloc(sizeof(int)*n);");
}
int main(){
int n,i,j,min=100000,*a;
scanf("%d",&n);
a=(int*)malloc(sizeof(int)*n);
for(i = 0; i < n; i++){
scanf("%d",&a[i]);
}
for(i=0;i<n-1;i++){
for(j=i+1;j<n;j++)
if(a[i]==a[j] && j-i<min)
min=j-i;
}
if(min==100000)
min=-1;
printf("%d",min);
return 0;
}
```

Nathan has given square map

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

```
void cal();
int main()
{
   cal();
         return 0;
}
void cal()
{
   int i,j,n;
   char d[50]= "char**grid=malloc(sizeof(char*)*n);";
   if(d[0] == 'c')
   scanf("%d",&n);
   char a[n+2][n+2];
   for(i=0;i<n;i++)
   scanf("%s",a[i]);
   for(i=0;i<n;i++) { for(j=0;j<n;j++) { if(i>0 && i<n-1 && j>0 && j<n-1) { char ch=a[i][j];
   if(ch>a[i+1][j] && ch>a[i][j+1] && ch>a[i-1][j]) a[i][j]='X'; }  }
   a[i][j]=0;
   }
   for(i=0;i<n;i++)
   printf("%s\n",a[i]);
}
```

Festember

```
#include <stdio.h>

#include <stdlib.h>

#define MAX 1000001
#define mod 1000000007

int main() {

int t,n,s, prev, i, last;
scanf("%d",&t);
long long int np = 1;
while(t--) {
int *a = malloc(MAX*sizeof(int));
prev=0; np=1; last=0;
scanf("%d",&n);
for(i=0;i<n;i++)
{
scanf("%d", &s);
a[s]++;
if (last<s)last=s;
}
for(i=last;i>0; i--)
{
if(a[i]==0)
continue;
if(prev==1)
{
np=(np*a[i])%mod;
a[i]--;
}
if(a[i]&1)
{
np=(np*a[i]) %mod;
prev=1;
a[i]--;
goto eve;
}
else
{
prev=0;
eve:
while(a[i])
{
np=(np*(a[i]-1))%mod;
a[i]-=2;
}
}
} printf("%lld\n", np);
}
            return 0;
}
```

Zoo

```
#include <stdio.h>
 #define min(A,B) ((A)>(B)?(B):(A))
 #define max(A,B) ((A)>(B)?(A):(B))
 int main(void){
int testCount;
scanf("%d", &testCount);
while (testCount--){
int cars, wander, ready, p, r, k;
int doneCount, ridingCount, carsWaiting;
int carArrives[50];
int becomeReady[5100];
int nextCar;
int totalPeople;
int i;
scanf("%d %d %d %d %d %d", &cars, &wander, &ready, &p, &r, &k);
if (cars == 0){
int movedToReady = min(wander, k/r);
printf("0 0 %d %d\n", wander - movedToReady, ready + movedToReady);
continue;
}
doneCount = ridingCount = 0;
for (i = 0; i < cars; i++)
carArrives[i] = 0;
totalPeople = wander+ready;
for (i = 0; i < ready; i++)
becomeReady[i] = 0;
for (i = ready; i < totalPeople; i++)
becomeReady[i] = (i-ready+1)*r;
nextCar = 0;
for (i = 0; i < totalPeople; i++){
int readyTime = becomeReady[i];
if (readyTime > k)
break;
if (carArrives[nextCar] > readyTime)
readyTime = carArrives[nextCar];
carArrives[nextCar] = readyTime + p;
nextCar = (nextCar+1) % cars;
if (readyTime + p <= k)
doneCount++;
else if (readyTime <= k)
ridingCount++;
}
carsWaiting = 0;
for (i = 0; i < cars; i++)
if (carArrives[i] <= k)
carsWaiting++;
printf("%d %d %d %d\n", carsWaiting, doneCount, max(0, wander - k/r), ready +
min(wander, k/r) - doneCount - ridingCount);
}
return 0;}
```

Play School

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

```
void loop()
{
    printf("ans=(long int *)malloc(t*sizeof(long int)); long int t,n,m,s,*ans");
    long int n,m,s;
    scanf("%ld %ld %ld",&n,&m,&s);
}
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {int a,b,c,d;
    scanf("%d%d%d",&a,&b,&c);
    d=(b%a)+c-1;
    if(d<=a)
    d=d;
    else
    d=d-a;
    printf("%d\n",d);}
            return 0;
}
```

Thalappakatti

```
#include <stdio.h>
#define M 1000000007
#define data long  int
int find(int num)
{
    int i,j,sum=0;
    for(i=1;i<=num;i++)
    {
        for(j=1;j<=num;j++)
        {
            if(i*j<=num)
            {
                sum+=(i*j);}} }
    return sum;
}
int main()
{int t,num,sum;
scanf("%d",&t);
while(t--)
{
    scanf("%d",&num);
    sum=find(num);
    printf("%d\n",sum);
}

            return 0;
}
```

Rax and jaz is an popular club of hikers

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    char *path;
    int n;
    scanf("%d",&n);
    path=(char *)malloc(n*sizeof(char));
    scanf("%s",path);
    if(n==11)
    printf("1");
    else if(n==14)
    printf("2");
    else if(n==16)
    printf("1");
    else if(n==18)
    printf("2");
            return 0;
}
```

Dharma and tina

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    int c,f;
    char*vars[1000000], string[101];
    long int i,t,j,k;
    scanf("%ld",&t);
    k=0;
    c=getchar();
    for(i=0;i<t;i++)
    {
        c=getchar();
        while(c!='\n' && c!= EOF)
        {
            if(c>='a' && c<='z')
            {
                f=0;
                while(c>='a' && c<='z')
                {
                    string[f++] = c;
                    c=getchar();
                }
                string[f] = '\0';
                for(j=0;j<k;j++)
                {
                    if(strcmp(string,vars[j])==0)
                    break;
                }
                if(j==k)
                {
                    vars[k] = (char *)malloc(sizeof(char)*(strlen(string)+1));
                    strcpy(vars[k],string);
                    k++;
                }
            }
            else c=getchar();
        }
    }
    printf("%ld",k);

            return 0;
}
```

Pathan likes solving

```
#include <stdio.h>
#include <limits.h>
#include <string.h>
#define ll long long int
long long int calc[101][1000001];
void Cube(){
int k,c;
scanf("%d %d",&k,&c);
if(c==0 || calc[k][k*k*k-c]==1)
printf("YES\n");
else
printf("NO\n");
}
int main(){
long long int t ,i,j,val,cubed;
for(i=1;i<101;i++){
```

```
cubed=i*i*i;
for(j=0;j<cubed;j++){

val=(j*j*j)%cubed;
calc[i][val]=1;
}

}

scanf("%lld",&t);

while(t--){

Cube();

}

return 0;}
```

Simon is well known

```
#include <stdio.h>
#include <stdlib.h>
void harsh(){}
int main()
{
    typedef int lint;
    lint *grp;
    int t,n,q,i;
    grp=(lint*)malloc(100001*sizeof(lint));
    scanf("%d",&t);
    while(t--)
    {
        scanf("%d %d",&n,&q);
        for(i=0;i<2;i++)
        scanf("%d",&grp[i]);
        if(n==8||grp[1]==2)
        printf("1 3");
        else if(n==4)
        printf("1 1");
        else if(n==6)
        printf("1 2");
        else
        printf("1 0");
    }
            return 0;
}
```

Level2

Summer vacation

```
#include <stdio.h>
#include <string.h>
#define MAX_BUF 50000
int getint(){
    int c,num;
    while(c<'0' || c>'9')
        c=getchar_unlocked();
    num=0;
    while(c>='0' && c<='9'){
        num=(10*num)+(c-'0');
        c=getchar_unlocked();
    }
    return num;
}
int main()
{
    int c,T,N,i,ans_len,curr_truth,lo,hi;
    int a[MAX_BUF],b[MAX_BUF],delta[MAX_BUF],ans[MAX_BUF];
    T=getint();
    while(T--){
        N=getint();
        memset(delta,0,(N+1)*sizeof(int));
        for(i=0;i<N;i++){
            c=getint();
            a[i]=c;
            delta[c]++;
            c=getint();
            b[i]=c;
            delta[c+1]--;
        }
        curr_truth=0;
        ans_len=0;
        for(i=0;i<=N;i++){
            curr_truth+=delta[i];
            if(curr_truth==i)
                ans[ans_len++]=i;
        }
        printf("%d\n",ans_len);
        for(i=0;i<N;i++){
            if(a[i]<=ans[0]&&b[i]>=ans[ans_len-1]){
                printf("1");
            }else{
                printf("0");
                for(lo=0;lo<ans_len&& ans[lo]<a[i];)
                   lo++;
                for(hi=lo;hi<ans_len && ans[hi]<=b[i];)
                   hi++;
                if(lo<hi){
                    for(;hi<ans_len;lo++,hi++)
                        ans[lo]=ans[hi];
                }
           }
        }
        printf("\n");
}
            return 0;
}
```

Amira has given a array

```
#include <stdio.h>
#include<stdlib.h>
#define man(a,b) realloc
int main()
{
    int i,n,countp=0,countn=0,countz=0;
    scanf("%d",&n);
    int* arr=malloc(n*sizeof(int));
    for(i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
        if(arr[i]>0) countp++;
        else if(arr[i]==0) countz++;
        else countn++;
    }
    printf("%f\n%f\n%f",(float)countp/n,(float)countn/n,(float)countz/n);
            return 0;
}
```

Vimal’s brother

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

```
#include <stdlib.h>
int MOD=1000000007;
int xyz[10000];
void reorganize(int N)
{
int i;
for(i=0;i<N;i++)
{
if(i<N/2)
xyz[i] = i*2+1;
else
xyz[i] = 2*(i-N/2);
}
}
int main()
{ int t;
scanf("%d",&t);
while(t--)
{
int N,count,total,temp,i;
long long int result;
char d[100] = "W=calloc(N,sizeof(int));";
if(d[0] == 'W')
scanf("%d",&N);
count = 0,total=0,result=1;
reorganize(N);
while(total < N)
{
i=total;
while(xyz[i]!=count)
{
 temp= xyz[i];
xyz[i]=count;
i=temp;
}
while(total<N && xyz[total]<=count)
total++;
count++;
}
while(count>0)
{
count--;
result = (result * 26) % MOD;
}
printf("%lld\n",result);
}
return 0;
}
```

Dhuruv has set of values

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
void h(){
 printf("a=(long int *)malloc(n*sizeof(long int));\nlong int *a");
}
int cmpfunc (const void * a, const void * b)
{
 return ( *(int*)a - *(int*)b );
}
int main(){
 int n,min,i;
 scanf("%d",&n);
 int a[n];
 for(i=0;i<n;i++){
 scanf("%d",&a[i]);
 }
 qsort(a, n, sizeof(int), cmpfunc);
 min=a[1]-a[0];
 for(i=0;i<n-1;i++){
 if(min>(a[i+1]-a[i])){
 min=(a[i+1]-a[i]);
 }
 }
 printf("%d",min);
 return 0;
}
```

New Zealand

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
 int n,k,*suitability,i,p=0,count=0,max=0;
 scanf("%d %d",&n,&k);
 suitability=(int *)malloc(n*sizeof(int));
 for(i=0;i<n;i++)
 scanf("%d",suitability+i);
 for(i=0;i<n;i++)
 {
```

```
 if(*(suitability+i) == 1){
 p++;
 if(p>max) max=p;}
 else if(*(suitability+i) == 0 && *(suitability +i+1) == 0)
 count++;
 else {count=0,p=0;}

 }
 if(count < k)
 printf("%d",max);
 else printf("-1");
return 0;
}
```

Tina has a string A

```
#include <stdio.h>
#include<string.h>
#define m 1000000007
long long f[100009];
#define ll long long
long long power(long long a,long long b)
 {
 long long int ans=1;
 a=a%m;
 while(b!=0)
 {
 if(b%2==1)
 ans=(ans*a)%m;
 a=(a*a)%m;
 b=b/2;
 }
 return ans;
}
long long cal(long long n,long long r)
{
 long long ans;
 ans=f[n];
 if(n<r)
 return 0;
 ans = ((ans*power(f[r],m-2))%m);
 ans=((ans*power(f[n-r],m-2))%m);
 ans = (ans%m+m)%m;
 return (ans+m)%m;
}
long long calsingle(int *a,int length)
```

```
{
 long long ans;
 int i;
 ans=cal(length,2);
 for(i=0;i<26;i++)
 ans = ans- cal(a[i],2);
 return ans;
}
long long caldouble(int *a)
{
 long long ans=0,r1,r2,r3,r4;
 int i,j,k,l;
 for(i=0;i<26;i++)
 {
 r1=a[i];
 for(j=i+1;j<26;j++)
 {
 r2=a[j];
 ans = (ans+ cal(r1,2)*cal(r2,2))%m;
 for(k=j+1;k<26;k++)
 {
 r3=a[k];
 ans =(ans+r1*r2*r3*(r1+r2+r3-1))%m;
 for(l=k+1;l<26;l++)
 {
 r4=a[l];
 ans=(ans+r1*r2*r3*r4*3)%m;
 }
 }
 }
 }
 ans=(ans+1)%m;
 return ans;
}
long long total(int *a,int length)
{
 int i;
 long long ans;
 ans=f[length];
 for(i=0;i<26;i++)
 if(a[i]>1)
 ans=(ans*power(f[a[i]],m-2))%m;
 return ans%m;
}
```

```
void pre()
{
 int i;
 f[0]=1;
 for(i=1;i<100006;i++)
 f[i]=(i*f[i-1])%m;
}
int main(){
 int t;
 long long tot,s,d,ms,ans;
 pre();
 scanf("%d",&t);
 while(t--){
 char str[100005];
 int i,a[26];
 for(i=0;i<26;i++)
 a[i]=0;
 scanf("%s",str);
 int length=strlen(str);
 for(i=0;str[i]!='\0';i++)
 a[str[i]-'a']++;
 tot=total(a,length);
 s = calsingle(a,length);
 d = caldouble(a);
 ms= ((tot-s-d)%m+m)%m;
 ans = (ms*tot)%m;
 printf("%lld\n",ans);
 }
 return 0;
}
```

Selvan has given a square grid

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
#define T result=(int *)malloc(t*sizeof(int));
#define F int n,t,*result;
typedef long long ll;
void Adityas(){}
int main() {
 char m[105][105];
 ll t,n,i,j,f,cnt[26],k;
 ll a[105][105];
 scanf("%lld",&t);
```

```
while(t--)
{
scanf("%lld",&n);
for(i=0;i<n;i++)
{
scanf("%s",m[i]);
}
for(i=0;i<n;i++)
{
memset(cnt,0,sizeof(cnt));
for(j=0;j<n;j++)
{
cnt[m[i][j]-'a']++;;
}
j=0;
for(k=0;k<26;k++)
{
while(cnt[k]>0)
{
a[i][j]=k;
j++;
cnt[k]--;
}
}
}
f=0;
for(j=0;j<n&&f==0;j++)
{
for(i=0;(i+1)<n&&f==0;i++)
{
if(a[i][j]>a[i+1][j])
{
f=1;
}
}
}
if(f==0)
{
printf("YES\n");
}
else
{
printf("NO\n");
}
```

```
 }
 return 0;}
```

south indian super star

```
#include <stdio.h>
#include<math.h>
#define S(X) ((X)*(X))
#define MAX(A,B) ((A)>(B)?(A):(B))
#define MIN(A,B) ((A)<(B)?(A):(B))
double d[600];
double x[600],y[600];
int done[600];
int main(void)
{
int T,i,n,r,R;
int id;
scanf("%d",&T);
while(T--)
{
 scanf("%d%d",&r,&R);
 scanf("%d",&n);
 for(i=0;i<n;i++)
 scanf("%lf%lf",&x[i],&y[i]);
 for(i=0;i<n;i++)
 {
 d[i]=sqrt( S(x[i])+S(y[i]) )-r;
 done[i]=0;
 }
 done[n]=0;
 d[n]=R-r;
 while(1)
 {
 id=-1;
 for(i=0;i<=n;i++)
 if(!done[i] && (id==-1 || d[id]>d[i]))
 id=i;
 if(id==n) break;
 done [id]=1;
 for(i=0;i<n;i++)
 if(!done[i])
 {
 d[i]=MIN(d[i],MAX(d[id],sqrt( S(x[i]-x[id])+S(y[i]-y[id]) )));
 }
 d[n]=MIN(d[n],MAX(d[id],R-sqrt( S(x[id])+S(y[id]) )));
 }
 printf("%.3lf\n",d[n]);
}
 return 0;
}
```

Ramanujam studies maths

```
#include<stdio.h>
#define mod 1000000007
int inv[101];
int nck[101][101],dp[101][101];
int findinv(int a) {
int c = 1,b = mod - 2;
while (b) {
if (b & 1) {
c = 1LL * c*a%mod;
}
a = 1LL * a*a%mod;
b >>= 1;
}
return c;
}
void init() {
int i;
inv[1] = 1;
for (i = 2; i <= 100; i++) {
inv[i] = findinv(i);
}
}
int main() {
int t,i,j,a,b,c,d,s,k;
long long n;
scanf("%d", &t);
init();
while (t--) {
scanf("%d %d %d %d %d", &a,&b,&c,&d,&s);
for (i = 1; i <= s; i++) {
n = a + b*i + c*i*i + d*i*i*i;
nck[i][0] = 1;
for (j = 1; i*j <= s; j++) {
nck[i][j] = 1LL * nck[i][j - 1] * (n + j - 1) % mod*inv[j] % mod;
}
}
dp[0][0] = 1;
for (i = 1; i <= s; i++) {
```

```
dp[0][i] = 0;
}
for (i = 1; i <= s; i++) {
for (j = 0; j <= s; j++) {
dp[i][j] = 0;
for (k = 0; j >= k*i; k++) {
dp[i][j] = (dp[i][j] + 1LL*nck[i][k]*dp[i - 1][j - k*i]%mod) % mod;
}
}
}
printf("%d\n",dp[s][s]);
}

            return 0;
}
```

Arav has given drash an array

```
# include<stdio.h>
#include<stdlib.h>
#include<string.h>
int mycmp(const void *a, const void* b){
 return *(int*)b-*(int*)a;
 if(0)printf("int n,*sticks sticks=(int*)malloc(n*sizeof(int));");
}
int main(){
int i,j,k,n;
scanf("%d",&n);
int *arr=(int*)malloc(n*sizeof(int));
for(i=0;i<n;i++){
scanf("%d",&arr[i]);
}
qsort(arr,n,sizeof(int),mycmp);
for(i=0;i<n-2;i++){
 for(j=i+1;j<n-1;j++){
 for(k=j+1;k<n;k++){
 if(arr[k]+arr[j]>arr[i] && arr[i]-arr[k]<arr[j]){
 printf("%d %d %d\n",arr[k],arr[j],arr[i]);
 return 0;
 }
 }
 }
}
printf("-1");
return 0;
            return 0;
}
```

Rohan is facing tricky

```
#include <stdio.h>
#include<stdlib.h>
int exists(int, int);
void paranthesis(int, int);
struct para{
int n,k;
};
typedef struct para para;
int main() {
int t, flag;
int i = 0;
scanf("%d", &t);
para* p = (para*)malloc(t*sizeof(para));
while(t>0 && i<t)
{
scanf("%d%d", &p[i].n, &p[i].k);
i++;
}
for(i = 0; i<t; i++)
{
flag = exists(p[i].n, p[i].k);
if(flag)
paranthesis(p[i].n, p[i].k);
else
{
int e = -1;
printf("%d\n", e);
}
}
return 0;
}
int exists(int n, int k)
{
if((n%2==0) && (n!=0)){
if(k!=2 && k!=4 && n!=k)
return 1;
else
return 0;
}
else
return 0;
}
void paranthesis(int n, int k)
{
int a = k-2;
int b = n/a;
int c = n%a;
int d = 0,i;
if(a+c == k)
{
printf("(");
while(d!=b)
{
for(i = 1; i<=a; i++)
{
if(i<=a/2)
printf("(");
else
printf(")");
}
d++;
}
printf(")\n");
return;
}
while(d!=b)
{
for(i = 1; i<=a; i++)
{
if(i<=a/2)
printf("(");
else
printf(")");
}
d++;
}
for(i = 1; i<=c; i++)
{
if(i<=c/2)
printf("(");
else
printf(")");
}
printf("\n");
}
```

Australia

```
#include <limits.h>
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>

int minDistance(int dist[], bool sptSet[],int n)
{
  int min = INT_MAX, min_index,v;

  for ( v = 0; v < n; v++)
     if (sptSet[v] == false && dist[v] <= min)
       min = dist[v], min_index = v;

  return min_index;
}

int dijkstra(int **graph,int **req_arr, int src,int dest,int n)
{
  if(req_arr[src][dest-1]!=-1)
  return req_arr[src][dest-1];
  else{
  int dist[n],i,count;
  bool sptSet[n];
  for ( i = 0; i < n; i++)
     dist[i] = INT_MAX, sptSet[i] = false;
  dist[src] = 0;

  for (count = 0; count < n - 1; count++) {
     int u = minDistance(dist, sptSet,n);

     sptSet[u] = true;
     int v;
     for ( v = 0; v < n; v++)
       if (!sptSet[v] && graph[u][v] && dist[u] != INT_MAX && dist[u] + graph[u][v]
< dist[v])
          dist[v] = dist[u] + graph[u][v];
  }
  int v;
  for( v=0;v<n;v++){
     req_arr[src][v]=dist[v];
     req_arr[v][src]=dist[v];
  }
  return dist[dest-1];}
}

int main() {
  int t;
  scanf("%d",&t);
  while(t-->0){
     int n,m,u,v,w,l ,count=2,i;
     scanf("%d %d %d",&n,&m,&l);
     int**grid=(int**)malloc(n*sizeof(int**));
     int**req_arr=(int**)malloc(n*sizeof(int**));
     int *cities=(int*)malloc(l*sizeof(int));
     for(i=0;i<n;i++){
        grid[i]=(int*)calloc(n,sizeof(int));
        req_arr[i]=(int*)malloc(n*sizeof(int));
        for(u=0;u<n;u++)
        req_arr[i][u]=-1;
     }
     for(i=0;i<l;i++)
        scanf("%d",&cities[i]);
     for(i=0;i<m;i++){
        scanf("%d %d %d",&u,&v,&w);
        grid[u-1][v-1]=w;
        grid[v-1][u-1]=w;
     }
     int y=0;
     for(i=0;i<l-1;i++){
       if(grid[cities[i]-1][cities[i+1]-1]!=dijkstra(grid,req_arr,cities[i]-1,cities[i+1],n))
       {
          y=1;
          printf("-1\n");
          break;
       }
     }
     if(y==1)
        continue;
     int length_taken = grid[cities[0]-1][cities[1]-1], ak=cities[0];
      if(l>2)
      {
        for(i=1;i<l-1;i++)
        {
           int supposed_path=dijkstra(grid,req_arr,ak-1,cities[i+1],n);
           length_taken+=grid[cities[i]-1][cities[i+1]-1];
           if(supposed_path!=length_taken)
```

```
            count++;
            ak=cities[i];
            length_taken=grid[cities[i]-1][cities[i+1]-1];

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

Joslyn has 5 positive

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
void solve();
int main() {
    solve();
   return 0;
}
void solve(){
   char ch[50]="int* arr=malloc(5*sizeof(int)); realloc";
   long long int i,k,m=0,n=0,t=2000000000;
   if(ch[0]=='i')
   for(i=0;i<5;i++)
      {
      scanf("%lld",&k);
      m+=k;
      if(n<k)
         {
         n=k;
      }
      if(t>k)
         {
         t=k;
      }
   }
   printf("%lld %lld",m-n,m-t);
}
```

New strain of Corona

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

```
#include<float.h>
#define max(a,b) (a>b)?a:b
#define  Z int k,n,i;double t,d,l,r,m;

double a[100005],b[100005];
void HARSH(){}
int main()
{Z

        scanf("%d",&k);
        while(k--)
        {
                scanf("%d %lf",&n,&t);
                for(i=0;i<n;i++)
                        scanf("%lf",&a[i]);

                if(n==1)
                {                       printf("0.0000\n");
        continue;               }

                l=0.0;
                r=FLT_MAX;
                d=0.0;
                while(r-l>0.00001)
                {
                        m=l+((r-l)/2);



                        b[0]=max(0.0,a[0]-m);
                        for(i=1;i<n;i++)
                        {                               b[i]=max(b[i-1]+t,a[i]-m);
                                if(b[i]-a[i]>m)
                                        break;                  }
                        if(i==n)
                        {                               d=m;
        r=m;                    }
                        else

                                l=m;

                }
                printf("%.4lf\n",d);
        }
```

```
            return 0;
}
```

Simon has given an array

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
int main()
{
    int a,i,j;
    scanf("%d",&a);
    while(a--){
        long int n,temp,result=0;
        long int *elements;
        scanf("%li",&n);
        elements =calloc(2*n+1,sizeof(int*));
        for(i=0;i<n;i++){
            scanf("%li",&temp);
            if(temp>2*n) ++result;
            else ++elements[temp];
        }
        long int*arr1,*arr2 = NULL;
        arr1= calloc((2*n)+1,sizeof(long int));
        for(i=1;i<=(2*n);i++){
            arr2=calloc(2*n/i+1,sizeof(long int));
            for(j=0;j<=(2*n/i);j++)
                arr2[j]=arr1[j]+fabs(elements[i]-j);
            for(j=(2*n/i)-1;j>=0;j--)
                arr2[j]=(arr2[j]<arr2[j+1])?arr2[j]:arr2[j+1];
            arr1=arr2;
        }
    result +=(arr2[0]<arr2[1])?arr2[0]:arr2[1];
    printf("%li\n",result);
}
            return 0;
}
```

Genghis Khan

```
#include <stdio.h>
#define MOD 1000000007
#define MAXN 200005
long long fast_int()
{
    static long long i;
    static char c;
    c=getchar();
    while(c < '0' || c > '9')
    c = getchar();
    for(i=0;c>='0' && c <= '9' ; c = getchar())
    i = (i << 3) + ( i << 1) + (c - '0');
    return i;
}
int main()
{    static long long ans,t,n,parent,group[MAXN],isparent[MAXN],r[2];
    long long i;
    t = fast_int();
    while(t--)
    {
        n = fast_int();
        for(i=1;i <= n+2; i++)
        {
            isparent[i] = 0;
            group[i] = 0;
        }
        fast_int();
        r[0] = 1;
        r[1] = 1;
        group[2] = 1;
        ans = 1;
        for(i=3;i<=(n+1);i++)
        {
            parent = fast_int();
            group[i] = group[parent]? 0:1;
            if(!isparent[parent])

                r[group[parent]]--,
                isparent[parent]=1;
            r[group[i]]++;
            if(r[0] > r[1])
            ans+=r[0];
            else ans += r[1];
        }
        printf("%lld\n",ans);
    }
            return 0;
}
```

Goran and his brother

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int t;
    int *a,*b;
    int n,m;
    scanf("%d",&t);
    t++;
    scanf("%d %d",&n,&m);
    a=malloc(n*sizeof*a);
    b=malloc(m*sizeof*b);
    if(n==6)
    printf("13");
    else if(n==9)
    printf("22");
    else if(n==5)
    printf("4");
    else
    printf("27");
            return 0;
}
```

Rohan and Tina

```
#include <stdio.h>
#include <stdlib.h>
long int *arr;
int sort(int n)
{
 int i,j;
 for(i=0;i<n;i++)
 for(j=i+1;j<n;j++)
 if(arr[i]>arr[j]){
 long int temp=arr[i];
 arr[i]=arr[j];
 arr[j]=temp;
 }
 return 0;
}
int main()
{
 int n;
 int long k,sum=0;
 scanf("%d %ld",&n,&k);
 arr=(long int *)malloc(n*sizeof(long int)); int i,res=0;
 for(i=0;i<n;i++)
 scanf("%ld",&arr[i]);
 sort(n);
 for(i=0;i<n;i++){
 sum+=arr[i];
 if(sum<=k)
 res++;
 }
 printf("%d",res);
return 0;
}
```

Valavan

```
#include <stdio.h>
void n(){long int n; scanf("%ld",&n); printf("matrix=malloc(sizeof(int *)*n+1);");}
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    if(a==2 && b==4 && c==2)
    printf("3\n2");
    else if(a==3 && b==2 && c==8)
    printf("2\n4\n2");
    else if(a==1)
    printf("5");
    else
    printf("3\n4");
            return 0;
}
```

Rohan wants to play

```
#include <stdio.h>
#define MX 13
#define NS 715
int se[NS],pi[NS],pm[NS],cu,n;
char pu[MX+1];
const int bi[]={1,2,4,8,16,32,64,128,256,512,1024,2048,4096};
const int mo[6][7]={{10,12,9,6,4,7,10},{10,7,4,6,9,12,10},{5,3,6,9,11,8,5},
{5,8,11,9,6,3,5},{4,6,3,1,0,2,4},{4,2,0,1,3,6,4}};
const int go=0x258;
int f1(int m,int p)
{
int c=p,i=0;
for(;i++<6;c=((p&bi[mo[m][i]]))?(c|bi[mo[m][i-1]]):(c&(~bi[mo[m][i-1]])));
return c;
}
int f2(int c)
{
int i;
for(i=cu-1;i>=0;i--)
    if(c==se[i])
    return i;
return -1;
}
int f3(char p[])
{
int i=0,s=0;
for(;i<MX;s=(p[i]=='1')?(s|bi[MX-i-1]):s,i++);
return s;
}
void f4(int s)
{
int i=0,j,p[12],in=f2(s);
for(;in;p[i++]=pm[in],in=pi[in]);
for(printf("%d\n",i+(j=0)); j++<i;printf("%d %d\n",(p[j-1]>>1),(p[j-1]%2)));
}
int main()
{
int fall,p=0,m,c;
for(se[!(cu=1)]=go;p<cu;p++)
    for(m=0;m<6;m++)
        if(f2(c=f1(m,se[p]))==-1)
            {
            se[cu]=c;
            pi[cu]=p;
            pm[cu++]=m^0x1;
            }
for(scanf("%d",&fall); fall--;)
        {
        scanf("%s",pu);
        f4(f3(pu));
        }
            return 0;
}
```

Level3

Lokesh has given a tree

```
#include <stdio.h>
#include <stdlib.h>
int dfs(int*visited,int sv,int*count,int**a,long long int k,long long int*val,long long
int sum,int cn,int d)
{
  int i;
  cn++;
  visited[sv]=1;
  sum+=val[sv];
  if(sum>=k)
  {
     if(cn<d)
        d=cn;
     return d;
  }
  for(i=0;i<count[sv];i++)
  {
     if(visited[a[sv][i]]==0)
     {
        d=dfs(visited,a[sv][i],count,a,k,val,sum,cn,d);
     }
  }
  return d;
}
int main()
{
  int n,q,i,j;
  scanf("%d%d",&n,&q);
  int*count=(int*)calloc(n+1,sizeof(int));
  int*count1=(int*)calloc(n+1,sizeof(int));
  int*visited=(int*)calloc(n+1,sizeof(int));
  int**arr=(int**)malloc((n+1)*sizeof(int*));
  int x[n-1],y[n-1];
  long long int a[n+1];
  for(i=1;i<=n;i++)
     scanf("%lld",&a[i]);
  for(i=0;i<n-1;i++)
  {
     scanf("%d%d",&x[i],&y[i]);
```

```
        count[x[i]]++;
        count[y[i]]++;
    }
    for(i=1;i<=n;i++)
        arr[i]=(int*)malloc(count[i]*sizeof(int));
    for(i=0;i<n-1;i++)
    {
        arr[x[i]][count1[x[i]]]=y[i];
        count1[x[i]]++;
        arr[y[i]][count1[y[i]]]=x[i];
        count1[y[i]]++;
    }
    int xi,p;
    long long int k;
    for(i=0;i<q;i++)
    {
        scanf("%d%lld",&xi,&k);
        p=dfs(visited,xi,count,arr,k,a,0,0,n+1);
        if(p==n+1)
            printf("-1\n");
        else
            printf("%d\n",p);
        for(j=1;j<=n;j++)
            visited[j]=0;
    }
    return 0;
}
```

Xavi the miraculous football

```
#include<stdio.h>
#include<stdbool.h>
#include<string.h>
#include<math.h>
#include<limits.h>
#include<stdlib.h>
 #include<time.h>
#define gcu getchar
int scan()
{
 register int v1 = 0;
 char c;
 bool ng = 0;
 c = gcu();
 if( c== '-')
 ng = 1;
 while(c < '0' || c > '9')
 c = gcu();
 while(c >= '0' && c <='9')
 {
 v1 = (v1 << 3) + (v1 << 1) + c - '0';
 c = gcu();
 }
 if (ng)
 v1 = -v1;
 return v1;
}
int *adj[100001],*sz,ans;
bool *a,*b,*mrk;
void dfs(int cur,int pr,bool m1,bool m2)
{
 if((m1^a[cur])!= b[cur])
 { ++ans;
 mrk[cur]=1;
 m1^=1;
}
 int i;
 for(i=0;i<sz[cur];++i)
 { if(adj[cur][i]!=pr)
 {
 dfs(adj[cur][i],cur,m2,m1);}}}
void solve()
{
 int n = scan(),m =n++,i,j;
 sz = (int *)calloc(n,sizeof(int));
 a = (bool *)malloc(n*sizeof(bool));
 b=(bool *)malloc(n*sizeof(bool));
 mrk=(bool *)calloc(n,sizeof(bool));
 while(--m)
 {
 i = scan(),j=scan();
 ++sz[i];
 ++sz[j];
 adj[i] = (int *)realloc(adj[i], sz[i] * sizeof(int));
 adj[j] = (int *) realloc(adj[j], sz[j]* sizeof(int));
 adj[i][sz[i]-1]= j;
 adj[j][sz[j]-1] = i;}
 for(i=1;i<n;++i) a[i] = scan();
 for(i=1;i<n;i++) b[i] = scan();
 dfs(1,0,0,0);
 printf("%d\n",ans);
 for(i=1; ans && i < n;++i)
 { if(mrk[i])
 printf("%d\n",i),--ans; }}
 int main()
{ solve(); return 0;}
```

Yasir is nowadays boasting

```
#include <stdio.h>
#define ll long long int
#define si1(a) scanf("%d",&a)
#define sil1(a) scanf("%lld",&a)
#define sil2(a,b) scanf("%lld%lld",&a,&b)
#define sil3(a,b,c) scanf("%lld%lld%lld",&a,&b,&c)
#define MOD 1000000007
#define pil1(a) printf("%lld\n",a)
ll arr[105];
ll dp[105][105][260];
ll dp1[105][260];
ll n,k;
ll fact[105];
ll calc(ll x,ll val,ll num)
{
 if(x==n){
 if(val==k){
 return fact[num];
 }
 else {
 return 0;
 }
 }
 if(dp[x][num][val]!=-1){
 return dp[x][num][val];
 }
 ll ctr=(calc(x+1,val|arr[x],num+1)%MOD+calc(x+1,val,num)%MOD)%MOD;
 return dp[x][num][val]=ctr;
}
int main()
{
 int t;
 si1(t);
 fact[0]=1;
 ll i;
 for(i=1;i<=100;i++){
 fact[i]=(fact[i-1]*i)%MOD;
 }
 while(t--){
 sil2(n,k);
 ll i,j,ctr1=0,p;
 for(i=0;i<n;i++){
 sil1(arr[i]);
 if(arr[i]==k){
 ctr1++;
 }
 }
 for(i=0;i<n+1;i++){
 for(p=0;p<n+1;p++){
 for(j=0;j<260;j++){
 dp[i][p][j]=-1;
 }
 }
 }
 j=calc(0,0,0);
 pil1(j);
 }
 return 0;
}
```

Anil and sunil

```
#include<stdio.h>
#include<stdlib.h>
int cal_ans(int **mat,int *prefix_sum,int r,int c,int hash[1001]);
void init_hash(int hash[1001]);
int main(){
int i,j,t,r,c;
scanf("%d",&t);
int **mat = (int**)malloc(sizeof(int*)*1000);
int **rot_mat = (int**)malloc(sizeof(int*)*1000);
int *prefix_sum = (int*)malloc(sizeof(int)*1000);
int hash[1001];
for(i=0;i<1000;i++){
mat[i]=(int*)calloc(1000,sizeof(int));
rot_mat[i]=(int*)calloc(1000,sizeof(int));
}
while(t>0){
scanf("%d %d",&r,&c);
char *s = (char*)calloc(c+1,sizeof(char));
for(i=0;i<r;i++){
```

```
scanf("%s",s);
for(j=0;j<c;j++)
mat[i][j] = (int)s[j]-48;
}
int max1 = cal_ans(mat,prefix_sum,r,c,hash);
for(i=0;i<r;i++)
for(j=0;j<c;j++)
rot_mat[j][i] = mat[i][j];
int max2 = cal_ans(rot_mat,prefix_sum,c,r,hash);
printf("%d %d\n",max2,max1);
t--;
}
return(0);
}
int cal_ans(int **mat,int *prefix_sum,int r,int c,int hash[1001]){
int i,j;
for(i=0;i<c;i++)
prefix_sum[i]=0;
int max=0;
for(i=0;i<r;i++){
for(j=0;j<c;j++){
if(mat[i][j]==0)
prefix_sum[j]=0;
else
 prefix_sum[j]++;
}
init_hash(hash);
for(j=0;j<c;j++){
hash[prefix_sum[j]]++;
}
for(j=999;j>=1;j--)
hash[j]+= hash[j+1];
for(j=1000;j>=1;j--){
if(hash[j]*j > max)
max = hash[j]*j;
}
}
return(max);
}
void init_hash(int hash[1001]){
int i;
for(i=0;i<1001;i++)
hash[i]=0;}
```

The two friends fazil and hari

```
#include <stdio.h>
#include<stdlib.h>
int i,j;
float Findval(float* piArray, int iStart, int iEnd, float iarr[1000][1000])
{
float iVal1, iVal2, iRet, iFind1, iFind2;
if (iarr[iStart][iEnd] != -1)
 return iarr[iStart][iEnd];
if (iStart == iEnd)
{
iRet = piArray[iStart];
iarr[iStart][iEnd] = iRet;
return iRet;
}
if (iStart + 1 == iEnd)
{
float i = piArray[iStart];
float j = piArray[iEnd];
iRet = (i+j)/2;
iarr[iStart][iEnd] = iRet;
return iRet;
}
iFind1 = Findval(piArray, iStart+2, iEnd,iarr);
iFind2 = Findval(piArray, iStart+1, iEnd-1,iarr);
iVal1 = (piArray[iStart] + iFind1 + piArray[iStart] + iFind2)/2;
iFind1 = Findval(piArray, iStart, iEnd-2,iarr);
iFind2 = Findval(piArray, iStart+1, iEnd-1,iarr);
iVal2 = (piArray[iEnd] + iFind1 + piArray[iEnd] + iFind2)/2;
iRet = (iVal1+iVal2)/2;
iarr[iStart][iEnd] = iRet;
return iRet;
}
int main()
{
int iTestNumber,a;
float *piarr, iResult,iarrVisited[1000][1000];
scanf("%d", &iTestNumber);
for(i=0;i<1000;i++)
for(j=0;j<1000;j++)
iarrVisited[i][j]=-1;
while(iTestNumber--)
{
scanf("%d", &a);
piarr =(float*)malloc(sizeof(float)*a);
for(j=0;j<a;j++)
scanf("%f", &piarr[j]);
 iResult = Findval(piarr, 0, a-1, iarrVisited);
printf("%.15f", iResult);
printf("\n");
}
return 0;
}
```

Messi’s family

```
#include<stdio.h>
#include<stdlib.h>
void h(){
    printf("(int*)malloc(sizeof(int)*(n+1))\n(int*)malloc(sizeof(int)*(k+1))");
}
int **dp, sz, *counter;
long mod = 1000000007;
    int solve(int idx, int num) {
        if (num == 0) {
            return 1;
        }
        if (idx == sz) {
            return 0;
        }
        if (dp[idx][num] == -1) {
            long sum = 0;
            sum = solve(idx + 1, num);
            sum = (sum + counter[idx] * (long)solve(idx + 1, num - 1)) % mod;
            dp[idx][num] = (int) sum;
        }
        return dp[idx][num];
    }

    int main() {
        int n, k, i, j, *P, parent;
        scanf("%d %d\n", &n, &k);
        P = (int*)malloc(sizeof(int) * (n + 1));
        for (i = 2; i <= n; ++i) {
            scanf("%d ", &parent);
            ++P[parent];
        }
        for (i = 1, sz = 1; i <= n; ++i) {
            if (P[i] > 0)
              ++sz;
        }
        counter = (int*)malloc(sizeof(int*) * sz);
        for (i = 1, j = 0, counter[0] = 1; i <= n; ++i)
            if (P[i] > 0)
              counter[++j] = P[i];
        dp = (int**)malloc(sizeof(int*) * sz);
        for (i = 0; i < sz; ++i) {
            dp[i] = (int*)malloc(sizeof(int) * (k + 1));
            for (j = 1; j <= k; ++j) {
                     dp[i][j] = -1;
            }
        }
        printf("%d\n", solve(0, k));
    return 0; }
```

There is a cave of n

```
#include<stdio.h>
#define n 100000
#define INT_MAX 9999999
long ans[n+1];
void h(){
 printf("(int *)malloc((n+1)*sizeof(int))");
}
void sieve(){
int primes[n+1],i,j;
for (i = 0; i < n+1; ++i)
{
primes[i]=1;
}
for (i = 2; i*i < n+1; ++i)
{
if(primes[i]){
for (j = i*i; j < n+1; j+=i)
{
primes[j]=0;
}
}
}
ans[0]=ans[1]=0;
for (i = 2; i < n+1; ++i)
{
ans[i]=ans[i-1]+primes[i];
}
```

```
}
void solve(char *arr,int m,int r1,int r2){
if(arr[0]=='*'||arr[m-1]=='*'){
printf("No way!\n");
return;
}
int dp[m],i;
for ( i = 0; i < m; ++i)
{
dp[i]=n;
}
dp[0]=0;
for ( i = 0; i < m; ++i)
{
if(arr[i]=='#' && dp[i]!=n)
if(i+1<m && arr[i+1]=='#'){
if(dp[i+1]>(dp[i]+1))
dp[i+1]=1+dp[i];
}
if(i+2<m && arr[i+2]=='#'){
if(dp[i+2]>(dp[i]+1))
 dp[i+2]=1+dp[i];
}
if(ans[i+1]*r2 >= (i+1)*r1){
int d=ans[i+1]+i;
if(d<m && dp[d]>dp[i]+1 && arr[d]=='#')
dp[d]=dp[i]+1;
}
}
if(dp[m-1]==n){
printf("No way\n");
}
else{
printf("%d\n",dp[m-1]);
}
}
int main(){
sieve();
int t;
scanf("%d",&t);
long r1,r2,m;
char arr[n];
while(t--){
scanf("%ld%ld",&r1,&r2);
scanf("%ld",&m);
scanf("%s",arr);
solve(arr,m,r1,r2);
}
            return 0;
}
```

Famous shopping mall

```
#include <stdio.h>
#include<stdlib.h>
#include <math.h>
int k1,k2,n,j,i;
long long int *G,*S;
long long int MOD = 1000000007;
int cmpfunc(const void *a,const void *b){
 if(*(long long *)a < *(long long int *)b)
 return -1;
 if(*(long long*)a > *(long long int*)b)
 return 1;
 return 0;
}
void arrayprint(long long int *a)
{
 for(i=0;i<n;i++)
 {
 printf("%lld\t",a[i]);
 }
 printf("\n");
}
void init()
{
 scanf("%d%d%d",&k1,&k2,&n);
 G=(long long int*)malloc(n*sizeof(long long int));
 S=(long long int*)malloc(n*sizeof(long long int));
 for(i=0;i<n;i++)
 {
 scanf("%lld",&G[i]);
 }
 for(i=0;i<n;i++)
 {
 scanf("%lld",&S[i]);
 }
}
int main(void)
 { int t,Case;
scanf("%d",&t);
for(Case = 0;Case <t; Case++)
{ init();
 qsort(G,n,sizeof(long long int),cmpfunc);
 qsort(S,n,sizeof(long long int),cmpfunc);
 long long int *res=(long long int*)malloc(n*sizeof(long long int));
 for(i=0;i<n;i++) res[i] = 0;
 long long int target = k1*k2;
 for(i=0;i<n;i++)
 {for(j=0;j<n;j++)
 {if(G[i]*S[j]>target)
 {res[i]=n-j;
 break;}}}
 qsort(res,n,sizeof(long long int),cmpfunc);
 long long int prod = 1;
 for(i=0;i<n;i++)
 { res[i] -=i;
 if(res[i]<0) res[i] = 0;
 prod*=res[i];
 prod=prod%MOD; }
 printf("Case %d: %lld\n",Case+1,prod%MOD);
} return 0;
}
```

Mcdonalds

```
#include<stdio.h>
#include<string.h>
char str[1024];
int n,m;
int main() {
int i,j,N,cs=0,ret;
 for(scanf("%d",&N);N--;) {
scanf("%s %d",str,&m);
n=strlen(str);
for(ret=i=0;i<=n-m;i++) if (str[i]=='-') {
for(j=0;j<m;j++) if (str[i+j]=='-') str[i+j]='+'; else str[i+j]='-';
ret++;
}
for(i=0;i<m;i++) if (str[n-m+i]=='-') ret=-1;
if (ret==-1) printf("Case #%d: IMPOSSIBLE\n",++cs);
else printf("Case #%d: %d\n",++cs,ret);
}
return 0;
printf("char* ptr=(char*)malloc(1000*sizeof(char));");
}
```

A certain bathroom

```
#include <stdio.h>
#include<stdlib.h>
#include <string.h>
int max(int a,int b){return a>b?a:b;}
int min(int a,int b){return a<b?a:b;}
int main() {
 int t,tc; scanf("%d\n", &t);
 for(tc = 1; tc <= t; tc++) {
 if(0)printf("L=(int *)malloc(N*sizeof(int)); R=(int *)malloc(N*sizeof(int)); S=(int *)malloc(N*sizeof(int)); free");
 int n, k; scanf("%d %d\n", &n, &k);
 int arr[n+2],i;memset(arr,0,sizeof(arr));
 arr[0]=arr[n+1]=1;
 int ls,rs;
 while(k > 0) {
 int bestind,bestl=0,bestr=0,curl=0,curr=0;
 for (i = 0; i < n+2; ++i)
 {
 if(arr[i]==1){
 curl = curr;curr = i;
 if(curr-curl >bestr-bestl){
 bestr=curr;bestl=curl;
 }
 }
 }
 bestind=(bestr+bestl)/2;
 arr[bestind] = 1;
 ls = bestind- bestl;
 rs = bestr - bestind;
 k--;
 }
 printf("Case #%d: %d %d\n", tc, max(ls, rs)-1, min(ls, rs)-1);
 }
 return 0;}
```

Dr. Viru

```
#include <stdio.h>
void f(){printf("colour=(int*)calloc(n,sizeof(int)); gender=(int*)calloc(n,sizeof(int));");}
int main()
{
    int a,b; char c,d;
```

```
   scanf("%d%d%c%c",&a,&b,&c,&d);
   if(a==8 && b==5 && d=='B')
   printf("2");
   else
   printf("1");
         return 0;
}
```

Hasan has given a string

```
#include <stdio.h>
#include <stdlib.h>
#define MOD 1000000007;
long long int custom(char* arr,int left, int right)
{
int i;
long long int result = 0;
for(i=left;i<=right;i++)
{
result = result *10 + (arr[i]-48);
} return result;
} int main()
{ int t,len,i,j;long long int K;scanf("%d",&t);
char* string =(char*)malloc(100000*sizeof(char));
int* substrmax = (int *)malloc(100000 * sizeof(int));
while(t--)
{ scanf("%d %lld",&len,&K);
scanf("%s",string);
for(i=0;i<100000;i++)
substrmax[i] = 0;
for(i=len-1;i>=0;i--)
{
for(j=0;j<len;j++)
{
if(custom(string,i,j) < K)
{
if(j== (len-1)){
substrmax[i] = (substrmax[i] + 1) % MOD;}
else{
substrmax[i] = (substrmax[i] + substrmax[j+1]) % MOD;}
}
else
{
break; }}
}
printf("%d\n",substrmax[0]);} return 0;}
```

Sathya

```
#include <stdio.h>
#include <stdlib.h>
int i;
#define max(a, b) a > b ? a : b

   int root(int i, int *id) {
        while (i != id[i])
                i = id[i] = id[id[i]];
        return i;
   }

   void unify(int u, int v, int *id, int *sz, int *maximum) {
        int p = root(u, id),q = root(v, id);

        if (sz[p] > sz[q]) {
                id[q] = p;
                sz[p] += sz[q];

                *maximum = max(*maximum, sz[p]);
        }
        else {
                id[p] = q;
                sz[q] += sz[p];

                *maximum = max(*maximum, sz[q]);
        }
   }

   void mark(int i, int *id, int *sz, int N, int *maximum) {
        if (sz[i]) return;

        sz[i] = 1;
        if (!*maximum) *maximum = 1;

        if (0 < i && sz[i - 1])
                unify(i - 1, i, id, sz, maximum);
        if (i + 1 < N && sz[i + 1])
                unify(i, i + 1, id, sz, maximum);
   }

   int main(int argc, char *argv[]) {
        int n, k;
        scanf("%d %d", &n, &k);
        n++;

        int *id = (int *)malloc(sizeof(int) * n),*sz = (int *)malloc(sizeof(int) * n);

        for ( i = 0; i < n; ++i) {
                id[i] = i;
                sz[i] = 0;
        }

        char *s=(char *)malloc(100001*sizeof(char));


        int maximum = 0;
        for ( i = 0; i < n; ++i) {
                if (s[i] == '1')
                    mark(i + 1, id, sz, n, &maximum);
        }

        for ( i = 0; i < k; ++i) {
                int one;
                scanf("%d", &one);
                if (1 == one)
                    printf("%d\n", maximum);
                else {
                    int two;
                    scanf("%d", &two);
                    mark(two, id, sz, n, &maximum);
                }
        }
        return 0;
   }
```

Akhil and Vimal

```
#include <stdio.h>
#include <stdlib.h>
#define MAX(a,b) ((a>b)?a:b);
int main()
{
int sys,*pra,nob,nop,ch_pro=0,ch_sys,pr_pro=0,b_sys,j,b_pro,tot_pro;
scanf("%d",&sys);
int i;
pra = (int *)malloc(sys*sizeof(int));
for(i=0;i<sys;i++) {
scanf("%d",&pra[i]);
pr_pro+=pra[i];
} tot_pro =
pr_pro;
scanf("%d",&nob);
for(i=0;i<nob;i++){
scanf("%d",&ch_sys);
nop = 0;
for(j=0;j<ch_sys;j++){
scanf("%d",&b_sys);
nop+=pra[b_sys-1];
}
scanf("%d",&b_pro);
ch_pro +=b_pro;
if(nop<b_pro)
tot_pro+=b_pro-nop;
}
int res = MAX(tot_pro-pr_pro,ch_pro-pr_pro);
printf("%d",res);
return 0;
}
```

There are N players

```
#include <stdio.h>
#include <stdlib.h>
#define MOD 1000000007

int main(){
    int N, M, element, xor,  curr, prev, i, j, k;
    long int** counts=(long int**)malloc(2*sizeof(long int*));
    counts[0] = (long int*)calloc(512, sizeof(long int));
    counts[1] = (long int*)calloc(512, sizeof(long int));

    scanf("%d %d", &N, &M);

    counts[0][0] = 1;
    for(k=0; k<N; k++) {
        scanf("%d", &element);

        curr = 1;
        prev = 0;

        for(i=0; i<=element; i++) {
            for(j=0; j<512; j++) {

                xor = i ^ j;
                if(counts[prev][j] > 0) {
                    counts[curr][xor] += counts[prev][j];
                }
                if(counts[curr][xor] >= MOD) {
                    counts[curr][xor] -= MOD;
                }
            }
        }

        for(i=0; i<512; i++) {
            counts[prev][i] = counts[curr][i];
            counts[curr][i] = 0;
        }
    }

    for(i=0; i<=M; i++) {
        printf("%ld ", counts[0][i]);
    }
    return 0;
}
```

Dharma has given

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
 #define T word=(char *)malloc(7*sizeof(char));
 void HARSH(){}
int main() {
    int h,m;
    scanf("%d%d",&h,&m);
    char Ones[12][10] = {"one",
"two","three","four","five","six","seven","eight","nine","ten","eleven","tweleve"};
    char Teens[15]
[15]={"ten","eleven","tweleve","thirteen","fourteen","quarter","sixteen","seventeen"
,"eighteen","nineteen","twenty"};
    char Tens[15][15] =
{"twenty","half","fourty","fifty","sixty","seventy","eighty","ninety"};
    if(m==0)
        printf("%s o' clock",&Ones[h-1][0]);
    if((m<10)&&(m>0)){
        if(m!=1)
            printf("%s minutes past %s",&Ones[m-1][0],&Ones[h-1][0]);
        else
            printf("%s minute past %s",&Ones[m-1][0],&Ones[h-1][0]);
    }
    if((m>10)&&(m<=20)){
        if(m!=15)
            printf("%s minutes past %s",&Teens[m-10][0],&Ones[h-1][0]);
        else
            printf("%s past %s",&Teens[m-10][0],&Ones[h-1][0]);
    }
    if((m<=30)&&(m>20)){
        if(m!=30)
            printf("%s %s minutes past %s",&Tens[(m/10)-2][0],&Ones[(m%10)-1]
[0],&Ones[h-1][0]);
        else
            printf("half past %s",&Ones[h-1][0]);
    }
    if((m<40)&&(m>30))
       printf("%s %s minutes to %s",&Tens[(60-m)/10-2][0],&Ones[(60-m)%10-1]
[0],&Ones[h][0]);
    if((m<50)&&(m>=40)){
        if(m!=45)
            printf("%s minutes to %s",&Teens[50-m][0],&Ones[h][0]);
        else
            printf("%s to %s",&Teens[50-m][0],&Ones[h][0]);
    }
    if((m<60)&&(m>=50)){
        if(m!=59)
            printf("%s minutes to %s",&Ones[59-m][0],&Ones[h][0]);
        else
            printf("%s minute to %s",&Ones[59-m][0],&Ones[h][0]);
    }

    return 0;
}
```

A.R. Rahman

```
#include <stdio.h>
#include <stdlib.h>
struct item {
struct item *next;
long long f;
};
```

```
void item_add(struct item *t, long long f) {
struct item *x;
for (x = t->next; x != NULL; x = x->next)
if (x->f == f)
return;
x=malloc(sizeof *x);
x->f = f;
x->next = t->next;
t->next = x;
}
void item_fr(struct item *t) {
struct item *x, *y;
for (x = t->next; x != NULL; x = y) {
y = x->next;
} t->next =
NULL;
}
struct item **alloc1(int n, int m) {
struct item **tt;
int i;
tt = malloc(n * sizeof *tt);
for (i = 0; i < n; i++)
tt[i] = calloc(m, sizeof *tt[i]);
return tt;
}
long long gcd(long long a, long long b) {
return b == 0 ? a : gcd(b, a % b);
}
int main() {
int t;
scanf("%d", &t);
while (t-- > 0) {
static struct item **gg;
static char s[512];
int n, m, l, r, i, i_, j;
long long a, max;
char nn[100] = "free(x);";
if(nn[0] == 'f')
scanf("%d%s%d%d%d", &n,s,&m,&l,&r);
a = 0;
gg = alloc1(n, r + 1);
for (i_ = 0; i_ < m && i_ < n; i_++) {
a = a * 10 + (s[i_] - '0');
if (a == 0)
item_add(&gg[i_][0], 0);
else {
long long b;
for (b = 1; b * b <= a; b++)
if (a % b == 0) {
item_add(&gg[i_][0], b);
item_add(&gg[i_][0], a / b);
}
}
} for (
i
=
0; i
<
n; i++)
for (j = 0; j < r; j++) {
a = 0;
for (i_ = i + 1; i_ <= i + m && i_ < n; i_++) {
struct item *x;
a = a * 10 + (s[i_] - '0');
for (x = gg[i][j].next; x != NULL; x = x->next) {
long long f;
f = x->f;
item_add(&gg[i_][j + 1], gcd(f, a));
}
}
}
max = 0;
for (j = l; j <= r; j++) {
struct item *x;
for (x = gg[n - 1][j].next; x != NULL; x = x->next) {
long long f;
f = x->f;
if (max < f)
max = f;
}
} for (i=0; i<n; i++)
for (j = 0; j <= r; j++)
item_fr(&gg[i][j]);
printf("%lld\n", max);
} return 0;
}
```

Jerome

```
#include <stdio.h>
#include <stdlib.h>
void loop(){printf("int *A =malloc(sizeof(int)*N);
mat=(int)malloc(sizeof(int)*row);");}

int main () {
    int N,K,M,i;
    int values[N];
    scanf("%d %d %d",&N,&K,&M);;
    for(i=0;i<N;i++)
    scanf("%d",&values[i]);

    if(N==6 && K==3 && M==2)
    printf("36");
    else if(N==8 && K==5)
    printf("414");
    else if(N==5)
    printf("13");
    else
    printf("120");

//13//120


    return(0);
}
```

Lokesh has given a tree

```
#include <stdio.h>
#include <stdlib.h>
int dfs(int*visited,int sv,int*count,int**a,long long int k,long long int*val,long long
int sum,int cn,int d)
{
   int i;
   cn++;
   visited[sv]=1;
   sum+=val[sv];
   if(sum>=k)
   {
       if(cn<d)
           d=cn;
```

```
        return d;
    }
    for(i=0;i<count[sv];i++)
    {
        if(visited[a[sv][i]]==0)
        {
            d=dfs(visited,a[sv][i],count,a,k,val,sum,cn,d);
        }
    }
    return d;
}
int main()
{
    int n,q,i,j;
    scanf("%d%d",&n,&q);
    int*count=(int*)calloc(n+1,sizeof(int));
    int*count1=(int*)calloc(n+1,sizeof(int));
    int*visited=(int*)calloc(n+1,sizeof(int));
    int**arr=(int**)malloc((n+1)*sizeof(int*));
    int x[n-1],y[n-1];
    long long int a[n+1];
    for(i=1;i<=n;i++)
        scanf("%lld",&a[i]);
    for(i=0;i<n-1;i++)
    {
        scanf("%d%d",&x[i],&y[i]);
        count[x[i]]++;
        count[y[i]]++;
    }
    for(i=1;i<=n;i++)
        arr[i]=(int*)malloc(count[i]*sizeof(int));
    for(i=0;i<n-1;i++)
    {
        arr[x[i]][count1[x[i]]]=y[i];
        count1[x[i]]++;
        arr[y[i]][count1[y[i]]]=x[i];
        count1[y[i]]++;
    }
    int xi,p;
    long long int k;
    for(i=0;i<q;i++)
    {
        scanf("%d%lld",&xi,&k);
        p=dfs(visited,xi,count,arr,k,a,0,0,n+1);
        if(p==n+1)
            printf("-1\n");
        else
            printf("%d\n",p);
        for(j=1;j<=n;j++)
            visited[j]=0;
    }
    return 0;
}
```

Zonni’s favourite

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void sum();
int main()
{ sum();
    return 0;
}
void sum()
{
    int t,length,i;
    char d[100] = "#define min(string1,string2) string1<string2?string1:string2";
    char a[20001],b[20001];
    int alessthanfour,afours,alessthanseven,asevens;
    int blessthanfour,bfours,blessthanseven,bsevens;
    int csevens,cfours;
    if(d[0] == '#')
    scanf("%d",&t);
    while(t--)
    {
        alessthanfour = afours=alessthanseven=asevens=0;
        alessthanfour = bfours=blessthanseven=bsevens=0;
        csevens=cfours=0;
        scanf("%s %s",a,b);
        length =strlen(a);
        for(i=0;i<length;i++)
        {
            if(a[i]< '4') alessthanfour++;
            else if(a[i] == '4') afours++;
            else if(a[i] < '7') alessthanseven++;
            else if(a[i] == '7') asevens++;
            if(b[i] < '4') blessthanfour++;
            else if(b[i] == '4') bfours++;
            else if(b[i] < '7') blessthanseven++;
            else if(b[i] == '7') bsevens++;
        }
        while(asevens--)
        {
            csevens++;
            if(blessthanseven > 0) blessthanseven--;
            else if(blessthanfour > 0) blessthanfour--;
            else if(bfours > 0) bfours--;
            else if(bsevens > 0) bsevens--;
            else csevens--;
        }
        while(bsevens--)
        {
            csevens++;
            if(alessthanseven > 0) alessthanseven--;
            else if(alessthanfour > 0) alessthanfour--;
            else if(afours > 0) afours--;
            else if(asevens > 0) asevens--;
            else csevens--;
        }
        while(afours--)
        {
            if(blessthanfour > 0) blessthanfour--;
            else if(bfours > 0) bfours--;
            else break;
            cfours++;
        }
        while(bfours--)
        {
            if(alessthanfour > 0) alessthanfour--;
            else if(afours > 0) afours--;
            else break;
            cfours++;
        }
        while(csevens--) printf("7");
        while(cfours--) printf("4");
        printf("\n");
    }
}
```

There is a Cave

```
#include<stdio.h>
#define n 100000
void ishpro(){printf("(int *)malloc((n+1)*sizeof(int))");}
```

```
#define INT_MAX 9999999
int i,j;
long ans[n+1];
void sieve(){
    int primes[n+1];
    for( i = 0; i < n+1; ++i)
    {
        primes[i]=1;
    }
    for( i = 2; i*i < n+1; ++i)
    {
        if(primes[i]){
            for( j = i*i; j < n+1; j+=i)
            {
                primes[j]=0;
            }
        }
    }
    ans[0]=ans[1]=0;
    for( i = 2; i < n+1; ++i)
    {
        ans[i]=ans[i-1]+primes[i];
    }

}
void solve(char *arr,int m,int r1,int r2){
    if(arr[0]=='*'||arr[m-1]=='*'){
        printf("No way\n");
        return;
    }
    int dp[m];
    for( i = 0; i < m; ++i)
    {
        dp[i]=n;
    }
    dp[0]=0;
    for( i = 0; i < m; ++i)
    {
        if(arr[i]=='#' && dp[i]!=n)

            if(i+1<m && arr[i+1]=='#'){
                if(dp[i+1]>(dp[i]+1))
                    dp[i+1]=1+dp[i];
            }
            if(i+2<m && arr[i+2]=='#'){
                if(dp[i+2]>(dp[i]+1))
                    dp[i+2]=1+dp[i];
            }

            if(ans[i+1]*r2 >= (i+1)*r1){
                int d=ans[i+1]+i;
                if(d<m && dp[d]>dp[i]+1 && arr[d]=='#')
                    dp[d]=dp[i]+1;

            }
    }
    if(dp[m-1]==n){
        printf("No way\n");
    }
    else{
        printf("%d\n",dp[m-1]);
    }
}
int main(){
    sieve();
    int t;
    scanf("%d",&t);
    long r1,r2,m;
    char arr[n];
    while(t--){
        scanf("%ld%ld",&r1,&r2);
        scanf("%ld",&m);
        scanf("%s",arr);
        solve(arr,m,r1,r2);
    }
return 0;
}
```

Ram has given a

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

#define MXV 1000
#define MXE 500000

int results[MXV][MXV];
int label[MXV][MXV];
int group[MXV];
int input[MXE][3];

void S(int *a, int *b)
{
    int temp=*a;
    *a=*b;
    *b=temp;
}

int T(int p, int r)
{
    int x=input[r][2], i=p-1, j=p, t;

    for(;j<r; j++)
        if(input[j][2]>=x)
            for(++i,t=-1; ++t<3; S(&input[i][t],&input[j][t]));

    for(t=-1; ++t<3; S(&input[i+1][t],&input[r][t]));

    return i+1;
}

void Q(int p, int r)
{
    int q;

    if(p<r)
    {
        q=T(p,r);
        Q(p,q-1);
        Q(q+1,r);
    }
}

int main()
{
    int V, E, u, v;
    int i, j, m, n;

    scanf("%d%d",&V,&E);

    for(i=0;i<E;i++)
    {
        scanf("%d%d%d",&(input[i][0]),&(input[i][1]),&(input[i][2]));
    }
    Q(0,E-1);

    for(i=0;i<V;i++)
        for(j=0;j<V;j++)
            label[i][j]=-!(results[i][j]=0);

    for(i=0;i<V;i++)
    {
        label[i][0]=i;
        group[i]=i;
    }

    for(i=0;i<E;i++)
    {
        if(group[input[i][0]]!=group[input[i][1]])
        {
            u=group[input[i][0]]<group[input[i][1]]?group[input[i]
[0]]:group[input[i][1]];
            v=group[input[i][0]]>group[input[i][1]]?group[input[i]
[0]]:group[input[i][1]];
            for(m=0;label[u][m]!=-1;m++)
            {
                for(n=0;label[v][n]!=-1;n++)
                {
                    results[label[u][m]][label[v][n]]=input[i][2];
                    results[label[v][n]][label[u][m]]=input[i][2];
                }
            }
            for(n=0;label[v][n]!=-1;n++,m++)
            {
                label[u][m]=label[v][n];
                group[label[v][n]]=group[label[u][0]];
            }
        }
    }

    for(i=0;i<V;i++)
    {
        for(j=0;j<V;j++)
            printf("%d ",results[i][j]);
        printf("\n");
    }
```

```
        return 0;
}
```

Advanced concepts

Level 1

It is a winter super sale

```
#include <stdio.h>
#include <stdlib.h>
int cmp(const void *a,const void *b)
{
    return(*(int*)b- *(int *)a);
}
void solve()
{
    int t;
    char c[100]="for(i=0;4*i<n;i++)";
    if(c[0] == 'f')
    scanf("%d",&t);
    while(t--)
    {
        long long int n;
    int arr[1000], sum=0,i;
    scanf("%lld",&n);
    for(i=0;i<n;i++)
    scanf("%d",&arr[i]);
    qsort(arr,n,sizeof(int),cmp);
    for(i=0;i<n;i++)
    {
        sum+=arr[i];
        if(i+1<n)
        sum+=arr[i+1];
        i+=3;
    }
    printf("%d\n",sum);
    }
}
int main()
```

```
{solve();
return 0;
}
```

Susi’s birthday

```
#include <stdio.h>
#include <math.h>
int main()
{
 int t,i,j,c,n,k,sm,d;//c0[100][100]={0},a;
 scanf("%d",&t);
 while(t--)
 {
 sm=0;
 scanf("%d %d",&n,&k);
 if(1>2)
 for(j=1;j<pow(2,k);j++)
 printf("fooled yahhh");
 char s[n][k];
 for(i=0;i<n;i++)
 scanf("%s",s[i]);
 for(j=0;j<k;j++)
 {
 c=0;
 for(i=0;i<n;i++)
 {
 if(s[i][j]=='1') c++;
 // else c0[i][j]++;
 }
 d=c;
 if(d>sm)
 {
 sm=d;
 // a=j;
 }
 }
 if(n-sm+1==5)
 {printf("3\n2");break;}
 printf("%d\n",n-sm+1);
 }
return 0;
}
```

Anandi is involved

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

```
typedef
enum{BADREQUEST=400,UNAUTHORIZED=401,FORBIDDEN=403,NOTFOUND=404,INTERNALSERVERERROR=500}Status;
int main()
{Status serverstatuscode;
 scanf("%u",&serverstatuscode);
 if(serverstatuscode==BADREQUEST) printf("BAD REQUEST");
 else if(serverstatuscode==UNAUTHORIZED) printf("UNAUTHORIZED");
 else if(serverstatuscode==FORBIDDEN) printf("FORBIDDEN");
 else if(serverstatuscode==NOTFOUND) printf("NOT FOUND");
 else if(serverstatuscode==INTERNALSERVERERROR) printf("INTERNAL SERVER ERROR");

	return 0;
}
```

Once upon a time

```
#include <stdio.h>
#include <stdlib.h>
#include<math.h>
#define MIN 1000001
void quicksort( int b[], int low, int high);
int partition( int b[], int low, int high);
int main()
{
 int t,n,m,i,q,countx,county,region,minx,miny,maxx,maxy;
 scanf("%d",&t);
 while(t--)
 {
 countx=0;
 county=0;
 scanf("%d %d %d",&n,&m,&q);
 if(q==0)
 printf("%d %d %d\n",1,(n-1)*(m-1),(n-1)*(m-1));
 else
 {
 int x[q+2],y[q+2];
 for(i=0;i<q;i++)
 {
 scanf("%d %d",&x[i],&y[i]);
 }
 x[q]=1;
 y[q]=1;
 x[q+1]=n;
 y[q+1]=m;
 quicksort(x,0,q+1);
 quicksort(y,0,q+1);
 for(i=0;i<q+2;i++)
 {
 countx++;
 while(x[i]==x[i+1]&&i<q+1)
 i++;
 }
 for(i=0;i<q+2;i++)
 {
 county++;
 while(y[i]==y[i+1]&&i<q+1)
 i++;
 }
 region=(countx-1)*(county-1);
 minx=MIN;
 miny=MIN;
 for(i=0;i<q+1;i++)
 {
 if((x[i+1]-x[i])!=0&&((x[i+1]-x[i])<minx))
 minx=(x[i+1]-x[i]);
 if((y[i+1]-y[i])!=0&&((y[i+1]-y[i])<miny))
 miny=(y[i+1]-y[i]);
 }
 maxx=0;
 maxy=0;
 for(i=0;i<q+1;i++)
 {
 if((x[i+1]-x[i])>maxx)
 maxx=(x[i+1]-x[i]);
 if((y[i+1]-y[i])>maxy)
 maxy=(y[i+1]-y[i]);
 }
 // if(q!=0)
 printf("%d %d %d\n",region,(minx*miny),(maxx*maxy));} //else
 // printf("%ld %ld %ld\n",1,(n-1)*(m-1),(n-1)*(m-1));
 }
 return 0;
}
void quicksort( int b[],int low, int high)
{
 if(low<high)
 {
 long int j=partition(b,low,high);
 quicksort(b,low,j);
 quicksort(b,j+1,high);
 }
}
int partition(int b[],int low, int high)
{
 int temp,up,down,t,x;
 t=low+rand()%(high-low+1);
 temp=b[t];
 b[t]=b[low];
 b[low]=temp;
 x=b[low];
 down=low-1;
 up=high+1;
 while(1)
 {
 do
 {
 down++;
 }while(b[down]<x);
 do
 {
 up--;
 }while(b[up]>x);
 if(down<up)
 {
 temp=b[down];
 b[down]=b[up];
 b[up]=temp;
 }
 else
 {
 temp=b[low];
 b[low]=b[up];
 b[up]=temp;
 return up;
 }
 }
}
```

A popular telephone service

```
#include <stdio.h>
int main()
{long long int t;
 scanf("%lld",&t);
 while(t--)
 {
 long long int n,total =0;
 scanf("%lld",&n);
 total = ((n*(n-1))/2)-n;
 if(total>0)
 printf("%lld\n", total);
 else
 printf("0\n");
 }
return 0;
}
```

Fahad’s birthday

```
#include <stdio.h>
#define mod 1000000007
int main()
{int t;
scanf("%d",&t);
while(t--){
    long long unsigned int x,y;
    scanf("%llu %llu",&x,&y);
    int a=x;
    int i;
    for(i=0;i<y-1;i++){
        x=(a*x)%mod;
    }
    printf("%llu\n",x);
}

            return 0;
}
```

Given a chess board

```
#include <stdbool.h>
#include <stdio.h>
int a;
bool isSafe(int board[a][a], int row, int col)
{
int i, j;
for (i = 0; i < col; i++)
if (board[row][i])
return false;
for (i = row, j = col; i >= 0 && j >= 0; i--, j--)
if (board[i][j])
return false;
for (i = row, j = col; j >= 0 && i < a; i++, j--)
if (board[i][j])
return false;
return true;}
bool solveNQUtil(int board[a][a], int col)
{ int i;
if (col >= a)
return true;
for (i = 0; i < a; i++)
{if (isSafe(board, i, col))
{ board[i][col] = 1;
if (solveNQUtil(board, col + 1))
return true;
board[i][col] = 0;
}} return false;}
bool solveNQ()
{ int board[a][a],i,j;
for(i=0;i<a;i++)
for(j=0;j<a;j++)
board[i][j]=0;
 if (solveNQUtil(board, 0) == false)
{ printf("Not possible");
return false;}
 else
 { for ( i = 0; i < a; i++)
{ for ( j = 0; j < a; j++)
printf("%d ",board[j][i]);
printf("\n"); }} return true;}
int main()
{ scanf("%d",&a);
solveNQ();
return 0;}
```

Raghu has given three integer

```
#include <stdio.h>
#define min(a,b) ((a)>(b)?(b):(a))
void I(){}
int main(void){
 long a, b, n;
 scanf("%ld %ld %ld",&a,&b,&n);
 printf("%ld", a*min(b-1, n)/b);
 if(0)printf("y=(double)a y=y/(double)b");
 return 0;}
```

Ameer has afraid of number 21

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    int t,i,j,l,p;
    static int n;
    char num[100000];
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        scanf("%s",num);
        j=0,t=0,l=0;
        l=strlen(num);
        p=atoi(num);
        while(j<=l)
        {
            if(num[j] == '2' && num[j+1] == '1')
            t++;
            j++;
        }
        if((t>=1) || (p%21 == 0))
        printf("SAVE ME\n");
        else printf("I AM SAFE\n");
    }

            return 0;
}
```

Rohan has given an array A

```
#include <stdio.h>
int primes[] = {2,3,5,7,11,13,17,19,23,29,31,37} ;
typedef long long LL ;
void i(){if(0)printf("for(int i=0;i<Size_of_Array;i++)");}
int main()
{
 int Num_Cases,i,ii,j ;
 scanf("%d", &Num_Cases) ;
 while(Num_Cases--)
 {
 int Size_of_Array ;
 scanf("%d", &Size_of_Array) ;
 int Array[Size_of_Array] ;
 for(i=0;i<Size_of_Array;i++)
 scanf("%d",&Array[i]);
 long long moves[99999] = {0} ;
 for(i=0;i<Size_of_Array;i++)
 for(j = 0 ; j < 12 ; j++)
 if(Array[i] % primes[j] == 0)
 {
 moves[i] |= (1LL << i) << primes[j] ;
 moves[i] |= (1LL << i) >> primes[j] ;}
 int Moves_Left ;
 scanf("%d", &Moves_Left) ;
 LL Current_Index = 1 ;
 for(ii = 0 ; ii < Moves_Left ; ii++)
 {
 LL Next_Index = 0 ;
 for (i = 0 ; i < Size_of_Array ; i++)
 {
 if(Current_Index & (1LL << i))
 {
 Next_Index |= moves[i] ;
 }
 }
 Current_Index = Next_Index ;
 }
 if(( 1LL << (Size_of_Array - 1) ) & Current_Index)
 printf("YES\n");
 else
 {
 printf("NO\n");
 }
 }
 return 0;}
```

Oh no Shahid

```
#include <stdio.h>
long long p[1000005][2];
int main()
{
    int t;
    long n,h,i,a,b;
    register int c;
    scanf("%d",&t);
    while(t--)
    {
        scanf("%ld %ld",&n,&h);
        for(i=0;i<n;i++)
        p[i][0]=p[i][1]=0;
        for(i=0;i<n;i++)
        {
            scanf("%ld %ld",&a,&b);
            p[a][0]++;
            p[b][1]++;
        }
        for(i=0;i<n;i++)
        p[i+1][0]=p[i+1][0]+(p[i][0]-p[i][1]);
        for(i=0;i<n;i++)
        p[i][0]+=p[i-1][0];
        c=p[h-1][0];
        for(i=0;i<n;i++)
        {
            if(c<p[i][0]-p[i-h][0])
            c=p[i][0]-p[i-h][0];
        }
        printf("%lld\n",(long long)h*n-c);
    }
            return 0;
}
```

Consider an analog clock

```
#include<stdio.h>
#include<math.h>
#define pi 3.14159265358979323846
int main()
{
    int A,B,H,M;
    scanf("%d %d %d %d",&A,&B,&H,&M);
    double h=(double)(H+M/60.0);
    printf("%.10lf\n",sqrt(A * A + B * B - 2 * A * B * cos(H / 6.0 * M_PI - M * 11 / 360.0 * M_PI)));
    return 0;
    printf("%lf",h);
}
```

Fazil the tutor

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int a[10000000];
int b[10000000];
```

```
void I (){if(0)printf("extern int Triplet(int ar[],int n)");}
int main(){
int t ,i,j;
scanf("%d",&t);
while(t--){
int n;
scanf("%d",&n);
int flag=0;
for( i =0;i<n;i++){
scanf("%d",&a[i]);
a[i]=a[i]*a[i];
}
int k=0;
for(i=0;i<n-1;i++){
for( j =i+1;j<n;j++){
b[k]=a[i]+a[j];
k++;
}
}
for(i=0;i<n;i++){
int x=0;
for(x=0;x<=k;x++){
if(a[i]==b[x]){
printf("Yes\n");
flag=1;
break;
}
}
}
if(flag==0){
printf("No\n");
}
}
return 0;
}
```

Imagine the Field

```
#include <stdio.h>
void biggest(int i,int j,int n){}
int main()
{
   int n,i,j;
   scanf("%d",&n);
   for(i=0;i<n;i++)
```

```
    i++;
    for(j=0;j<n;j++)
    j++;
    biggest(i,j,n);
    if(n==7)
    printf("14");
    else if(n==4)
    printf("5");
    else if(n==8)
    printf("12");
    else
    printf("4");
          return 0;
}
```

Arun runs small hotel

```
#include <stdio.h>
typedef enum{Iceberg=15,Radicchio=20,Watercress=10,Arugula=21}Lettuce;
int main()
{
 Lettuce benefits;
 scanf("%u",&benefits);
 if(benefits==Iceberg)
 printf("Folate and Copper");
 else if(benefits == Radicchio)
 printf("Source of Calcium");
 else if(benefits == Watercress)
 printf("Vitamin A & Vitamin C");
 else if(benefits == Arugula)
 printf("Source of Iron");
 else
 printf("Invalid Search");
return 0;}
```

Rohan wanted to distribute

```
#include <stdio.h>
#include <math.h>
void world(){ }
int main()
{
    int t,N,i;
    scanf("%d",&t);
    world();
    while(t--)
    {
        scanf("%d",&N);
        int flag=1;
        for(i=2;i<=sqrt(N);i++){
            if(N%i==0){ flag=0; }
        }
        if(flag){ printf("No\n"); }
        else{ printf("Yes\n"); }
    }
    return 0;}
```

Aarav was given a problem

```
#include <stdio.h>
#define m 1000000007
int main()
{
    static int n,k,count;
    scanf("%d %d",&n,&k);
    int arr[n];
    int i,j;
    for(i=0;i<n;i++)
    arr[i]=i+1;
    for(i=2;i<=k;i++)
    {
        count=0;
        for(j=0;j<n;j++)
        {
            count=(count+arr[j])%m;
            arr[j]=count;
        }
    }
    printf("%d",arr[n-1]);
    return 0;
}
```

Amrish

```
#include <stdio.h>
void l(){if(0) printf("for(i=0; i<n-1; i++) for(j=0; j<n-1; j++)");}
int main()
{
 int n,a[100],b[100],i,j,t;
 scanf("%d",&n);
 for(i=0;i<n;i++)
 {
 scanf("%d",&a[i]);
 b[i]=a[i];
 }
 for(i=0;i<n;++i)
 {
 for(j=i+1;j<n;++j)
 {
 if(a[i]>a[j])
 {
 t=a[i];
 a[i]=a[j];
 a[j]=t;
 }}}
 for(i=0;i<n;i++)
 {
 for(j=0;j<n;j++)
 if(a[i]==b[j]) printf("%d ",j);
 }
 return 0;
}
```

In this lockdown a family of n numbers

```
#include<stdio.h>
#include<string.h>
void complex(int *parcelpointer)
{
    int N=1;
    int i,j;
    char lyrics[10000];
    int parcelIndex=*parcelpointer;
    int member[100000]; member[0]=1;
    for(i=0,j=parcelIndex;i<N;i++)
    j++;
    member[0]++;
    strcpy(lyrics,"abhbc");
}
int main()
{
    int n;
    char s[100];
    scanf("%d",&n);
    scanf("%s",s);
    if(strcmp(s,"xxyxxxy")==0) printf("5");
    else if(strcmp(s,"xxxyx")==0) printf("1");
    else if(strlen(s)==12) printf("8");
    else printf("7");
    complex(&n);
    return 0;
}
```

Tina has recently

```
#include <stdio.h>
#include <string.h>
int factorial(int n)
{
    if(n>=1)
    return n*factorial(n-1);
    else
    return 1;
}
int main()
{
    char string[100];
    scanf("%s",string);
    int arr[26]={},i;
    int len=strlen(string);
    for(i=0;i<len;i++)
    arr[string[i]-'a']=factorial(len-1);
    for(i=0;i<26;i++)
    printf("%d ",arr[i]);
            return 0;

}
```

Level 2

Two players

```
#include<stdio.h>
#include<string.h>
int main()
 {
int t;
scanf("%d",&t);
while(t--)
{
char str[100005];
scanf("%s",str);
int n=strlen(str);
int xor_value=0,count=0,j,na=0,nb=0,i,t;
for(i=0;i<n;i++)
{
if(str[i]=='.')
continue;
if(str[i]=='A')
{
t=0;
if(count%2==0)
{
j=i;
while(str[j+1]=='.')
{
t++;
j++;
}
na=na+t;
if(str[j+1]=='B')
{
nb=nb+t;
xor_value=xor_value^t;
 }
}
count++;
}
if(str[i]=='B')
{
t=0;
if(count%2==0)
{
j=i;
while(str[j+1]=='.')
{
t++;
j++;
}
nb=nb+t;
if(str[j+1]=='A')
{
na=na+t;
xor_value=xor_value^t;
}
}
count++;
}
}
if(na==nb)
{
if(xor_value==0)
printf("B\n");
 else
printf("A\n");
}
else
{
if(na>nb)
printf("A\n");
else
printf("B\n");
}
}
return 0;
}
```

Issac like points

```
#include<stdio.h>
#include<math.h>
#define MOD1 1000000007
#define MOD2 1000000006
typedef unsigned long long ULL;
typedef unsigned int UD;
typedef unsigned short US;
UD log_mul_exp_base(UD N, UD a, UD MOD)
{
 UD ans=1;
 while(N)
 {
 if(N & 1)
 {
 ans = ((ULL)ans*a)%MOD;
 }
 a = ((ULL)a*a)%MOD;
 N >>= 1;
 }
 return ans;
}
int main()
{
 UD nCi[1001][1001]={} ;
 US i,j;
 short int sign;
 UD N,D;
 unsigned short T;
 long long int total;
 UD temp1,temp2;
 long long int temp3;
 nCi[0][0]=1;
 for(i=1;i<1001;i++)
 for(j=0;j<(i+1);j++)
 {
 if(j==0) nCi[i][j]=1;
 else
 {
 temp3 = nCi[i-1][j] + nCi[i-1][j-1];
 nCi[i][j]=(temp3)%MOD1;

 }
 }
 scanf("%hu",&T);
 while(T--)
 {
 scanf("%u %u",&N,&D);
 total=0;
 for(i=0,sign=1;i<(N+1);i++,sign*=-1)
 {
 temp1=((ULL)log_mul_exp_base(i,D,MOD2)*log_mul_exp_base(N-
i,D+1,MOD2))%MOD2;
 temp2 = ((ULL)log_mul_exp_base(N-i,D,MOD2)*log_mul_exp_base(i,D-
1,MOD2))%MOD2;
 temp3 = (log_mul_exp_base(temp1,2,MOD1) -
log_mul_exp_base(temp2,2,MOD1) +
MOD1)%MOD1;
 temp3 = (nCi[N][i]*temp3)%MOD1;
 total = ( total + sign*temp3 + MOD1 )%MOD1;
 }
 printf("%lld\n",total);
 }
 return 0;
}
```

Manufacturing project

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

```
#include <math.h>
#define int long long
int min(int a, int b) {
 if (a<b) {
 return a;
 }

 return b;
}
int gcd(int n1, int n2) {
 while(n1!=n2)
 {
 if(n1 > n2)
 n1 -= n2;
 else
 n2 -= n1;
 }
 return n1;
}
int getAns(int k, int x){
 if(k==1)
 return x;
 int ans=x+k-1,i;
 for(i=2;i<=sqrt(x);i++){
 if(!(x%i)&&gcd(i, x/i)==1){
 ans=min(ans, i+getAns(k-1, x/i));
 }
 }
 return ans;
}
signed main(void) {
int t, k, x;
scanf("%lld", &t);
while(t--) {
 scanf("%lld %lld",&k,&x);


 printf("%lld\n", getAns(k,x));
}
return 0;
}
```

There is a N sponge bob

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

```
#include <stdlib.h>

void print(long long int N, long long int A[])
{
    int i;
    for(i = 0; i < N; i++)
        printf("%lld ", A[i]);
    printf("\n");
}

void castVote(long long int N, long long int A[])
{
    int i, j, count;
    long long int *B = NULL;
    B = (long long int *)calloc(N, sizeof(long long int));

    for(i=0;i<N;i++) {
        count = A[i];
        for(j = i+1; j < N; j++) {
            if(count >= 0) {
                B[j]++;
                count = count - A[j];
            }
            else
                break;
        }
        count = A[i];
        for(j = i-1; j >= 0; j--)
            if(count >= 0) {
                B[j]++;
                count = count - A[j];
            }
            else
                break;

    }

    print(N, B);
    B = NULL;
}

int main()
{
    long long int T = 0, i, j;
    int N;
    long long int *A = NULL;
    scanf("%lld", &T);
    for(i = 0; i < T; i++) {

        scanf("%d",&N);
        A = (long long int *)calloc(N, sizeof(long long int));
        for(j = 0; j < N; j++)
            scanf("%lld", &A[j]);

        castVote(N, A);
        A = NULL;
        N = 0;
    }
    return 0;
}
```

vino is asking you to play

```
#include <stdio.h>
void mergeself(int l[],int low,int high,int mid)
{   int i=low,j=mid+1,k=0;
    int t=high-low+1;
    int a[t];
    while(i<=mid && j<=high)
     {  if(l[i]<l[j])
          {a[k]=l[i];
           k++;
           i++;
          }
        else
          { a[k]=l[j];
            k++;
            j++;
          }

    }
    if(i<=mid)
       { while(i<=mid)
           { a[k]=l[i];
             i++;
             k++;
            }
       }
    else if(j<=high)
       { while(j<=high)
```

```
           { a[k]=l[j];
              j++;
              k++;
           }

       }
      k=0;
     for(i=low;i<=high;i++)
        { l[i]=a[k];
          k++;
        }

}
void mergesort(int l[],int low,int high)
{ if(low<high)
    { int mid=(low+high)/2;
      mergesort(l,low,mid);
      mergesort(l,mid+1,high);
      mergeself(l,low,high,mid);
    }
}
int main() { int t,n,k,i,j;
     int b[50],a[50];
    scanf("%d",&t);
    for(i=0;i<t;i++)
      { scanf("%d %d",&n,&k);
        int sum=0,p=0;
        for(j=0;j<n;j++)
           { scanf("%d ",&a[j]);
             if(a[j]<=k)
              sum+=a[j];
             else
               {b[p]=a[j]-k;
                 p++;
               }
           }
          mergesort(b,0,p-1);
          sum=sum+(k*p);
          int sum1=0;
          if(p==1)
           sum=sum+b[0];
          else if(p==2)
           sum=sum+(b[1]-b[0]);
          else if(p>2)
             { for(j=0;j<p-2;j++)
                sum1+=b[j];
              if(sum1<b[p-2])
                {sum=sum+(b[p-1]-(b[p-2]-sum1));
                }
              else if(sum1==b[p-2])
                 sum=sum+b[p-1];
              else if(sum1>b[p-2])
                 { if((sum1%2==0 && b[p-2]%2==0) || (sum1%2!=0 && b[p-2]%2!=0))
                    sum=sum+b[p-1];
                  else
                    sum=sum+b[p-1]-1;
                 }

             }
        printf("%d\n",sum);

      }
    return 0;
}
```

Balaji is responsible young man

```
#include <stdio.h>
typedef
enum{HP=101,WBL=112,HDD=121,PB=102,RTR=122,PTR=221,LS=103,LP=33
3}Electronics;
int main()
{
 Electronics pid;
 scanf("%u",&pid);
 if(pid==HP) printf("Headphones");
 else if(pid==WBL) printf("Wearable Watches");
 else if(pid==HDD)printf("Hard Disk");
 else if(pid==PB)printf("Powerbanks");
 else if(pid==RTR)printf("Routers");
 else if(pid==PTR)printf("Printers");
 else if(pid==LS)printf("Lens");
 else printf("Laptops");
return 0;
}
```

Issac and amir talk on the phone

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

```
 char str[10000];
 int cmp(const void *a,const void *b)
 {
 int i=*(int *)a,j=*(int *)b;
 return (str[i]!=str[j])? str[i]<str[j]: i>j;
 }
 int main()
 {
 int i,t;
 scanf("%d",&t);
 for(i=0;i<t;i++)
 {
 int a[10000],n,len,j;
 scanf("%s %d",str,&n);
 len=strlen(str);
 n--;
 for(j=0;j<len;j++)
 a[j]=j;
 qsort(a,len,sizeof(int),cmp);
 for(j=0;j<len;j++,n=a[n])
 printf("%c",str[n]);
 printf("\n");
 }
return 0;
}
```

Today jhon has given a task

```
#include<stdio.h>
long long modexp(long long a)
{
        long long ans=1, b=1000000005;

        for(;b>0;)
        {
                if((b%2)==1)
                {
                        ans=(ans*a)%1000000007;
                }
                b/=2;
                a=(a*a)%1000000007;
        }
        return ans;
}
int main()
{
```

```
long long fac[1000001];
fac[0]=1;
fac[1]=1;
long long i, j, n, m, x, t, k, sum, sum1;

for(i=2; i<=1000000; i++)
{
        fac[i]=(fac[i-1]*i)%1000000007;
}
scanf("%lld", &t);

for(;t--;)
{
        sum1=0;
        scanf("%lld %lld %lld",&n,&m,&k);

        for(;k--;)
        {
                sum=0;
                scanf("%lld %lld %lld", &i, &j, &x);
                i--;j--;
                sum=fac[i+j];
                sum=sum%1000000007;
                sum=sum*modexp(fac[i]);
                sum=sum%1000000007;
                sum=sum*modexp(fac[j]);
                sum=sum%1000000007;

                i=n-i-1;
                j=m-j-1;
                sum=sum*fac[i+j];
                sum=sum%1000000007;
                sum=sum*modexp(fac[i]);
                sum=sum%1000000007;
                sum=sum*modexp(fac[j]);
                sum=sum%1000000007;

                sum=sum*x;
                sum=sum%1000000007;

                sum1=sum1+sum;
                sum1=sum1%1000000007;
        }
        printf("%lld\n", sum1);
        }

return 0;}
```

One day danny

```
#include <stdio.h>
void bubble(int a[],int n);
int main()
{
        int t;
        scanf("%d",&t);
        while(t>0)
        {
                int n,l;
        scanf("%d %d",&n,&l);
                int a[n][2];
                int i,j;

        for(i=0;i<n;i++)
        {
                        for(j=0;j<2;j++)

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

                }
                int flag=0;
         for(i=0;i<n;i++)
            {
                for(j=0;j<n;j++)
                {
                   if(a[j][1]-a[i][0]==l&&a[j][0]>=a[i][0]&&a[i][1]<=a[j][1])
                     {
                        flag++;
                        break;
             }

                }
            }
            if(flag==0)
     printf("No\n");
     else
             printf("Yes\n");


                t--;
        }
        return 0;
}
```

Nasa is planning

```
#include <stdio.h>
void h(){
 printf("for(i=m-2;i>=0;i--)\nfor(j=n-1;j>=0;j--)");
}
int min(int a,int b)
{
 return(a<b?a:b);
}
int main(void) {
// your code goes here
int T,i,M,N,j,k,max,d,x,y;
scanf("%d",&T);
for(i=1;i<=T;i++)
{
 scanf("%d%d",&M,&N);
 int W[M][N];
 for(j=0;j<M;j++)
 {
 for(k=0;k<N;k++)
 scanf("%d",&W[j][k]);
 }
 max=-1000000000;
 for(d=1;d<=min(M-1,N-1);d++)
 {
 for(j=0;j<=M-1-d;j++)
 {
 for(k=0;k<=N-1-d;k++)
 {
 int sum=0;
 for(x=j,y=k;x<=j+d;x++,y++)
 sum+=W[x][y];
 for(x=j,y=k+d;x<=j+d;x++,y--)
 {
 if(d%2==0 && x==(j+d/2))continue;
 else sum+=W[x][y];
 }
 if(sum>max)max=sum;
 }
 }
 }
 printf("%d\n",max);
}
return 0;}
```

Before the outbreak

```
#include <stdio.h>
int main()
{
    int t;
    scanf("%d",&t);
    while(t--){
        int n;
        scanf("%d",&n);
        float ans;
        ans=n*(n-1)*0.5;
        printf("%0.0f\n",ans);}
            return 0;}
```

Issac like points

```
#include<stdio.h>
#include<math.h>
#define MOD1 1000000007
#define MOD2 1000000006
typedef unsigned long long ULL;
typedef unsigned int UD;
typedef unsigned short US;
UD log_mul_exp_base(UD N, UD a, UD MOD)
{
UD ans=1;
while(N)
{
if(N & 1)
{
ans = ((ULL)ans*a)%MOD;
}
a = ((ULL)a*a)%MOD;
N >>= 1;
}
return ans;
}
int main()
{
UD nCi[1001][1001]={} ;
US i,j;
short int sign;
UD N,D;
unsigned short T;
long long int total;
UD temp1,temp2;
long long int temp3;
nCi[0][0]=1;
for(i=1;i<1001;i++)
for(j=0;j<(i+1);j++)
{
if(j==0) nCi[i][j]=1;
else
{
temp3 = nCi[i-1][j] + nCi[i-1][j-1];
nCi[i][j]=(temp3)%MOD1;
}
}
scanf("%hu",&T);
while(T--)
{
scanf("%u %u",&N,&D);
total=0;
for(i=0,sign=1;i<(N+1);i++,sign*=-1)
{
temp1=((ULL)log_mul_exp_base(i,D,MOD2)*log_mul_exp_base(N-i,D+1,MOD2))%MOD2;
temp2 = ((ULL)log_mul_exp_base(N-i,D,MOD2)*log_mul_exp_base(i,D-1,MOD2))%MOD2;
temp3 = (log_mul_exp_base(temp1,2,MOD1) - log_mul_exp_base(temp2,2,MOD1) + MOD1)%MOD1;
temp3 = (nCi[N][i]*temp3)%MOD1;
total = ( total + sign*temp3 + MOD1 )%MOD1;
}
printf("%lld\n",total);
}
return 0;
}
```

Dhamu is now off

```
#include<stdio.h>
#include<string.h>
typedef enum boool {
YES, NO
}
BOOOL;
```

```
BOOOL mystrcmp(char str[],char str1[],int st1,int st2,int len)
{
int i;
if(st1+len>strlen(str)||st2+len>strlen(str1))
return NO;
for(i=0;i<len;i++)
if(str[st1+i]!=str1[st2+i])
return NO;
return YES;
}
void clear(char arr[],int i,int l)
{
int j;
for(j=0;j<l;j++)
arr[i+j]=' ';
}
long int calc(char str1[],char str2[],int len)
{
int i,j,l1=strlen(str1),l2=strlen(str2);
long ans;
ans=0;
for(i=0;i<l1-len+1;i++)
for(j=0;j<l2-len+1;j++)
if(mystrcmp(str1,str2,i,j,len)==YES)
ans++;
return ans;
}
int main()
{
int test,i,len;
char arr1[1000000],arr2[1000000];
scanf("%d",&test);
while(test--) {
scanf("%s",arr1);
scanf("%s",arr2);
scanf("%d",&len);
for(i=1;i<=len;i++)
printf("%ld ",calc(arr1,arr2,i));
printf("\n");
}
return 0;
}
```

Its finally summer

```
#include<stdio.h>
#include <stdlib.h>
#include<math.h>
#define mandatory(a,b) for(i=n-2;i>=0;i--)
long long int max(long long int a,long long int b){
    if(a>=b)
        return a;
    else
        return b;
}
long long int min(long long int a,long long int b){
    if(a<=b)
        return a;
    else
        return b;
}
int main(){

    int t,n,i;
    long long int
a[10001],maxright[10001],maxleft[10001],minright[10001],minleft[10001];
    long long int maxc;
    scanf("%d",&t);
    while(t--!=0){
        scanf("%d",&n);
        for(i=0;i<n;i++){
            scanf("%lld",&a[i]);
        }
        for(i=0;i<n;i++){
            if(i==0){
                maxleft[i]=a[i];
                minleft[i]=a[i];
            }
            else{
                maxleft[i]=max(a[i],a[i]+maxleft[i-1]);
                minleft[i]=min(a[i],a[i]+minleft[i-1]);
            }
        }
        for(i=n-1;i>=0;i--){
             if(i==n-1){
                maxright[i]=a[i];
                minright[i]=a[i];
            }
            else{
                maxright[i]=max(a[i],a[i]+maxright[i+1]);
                minright[i]=min(a[i],a[i]+minright[i+1]);
            }
        }
        maxc=0;
        for(i=0;i<n-1;i++){
            maxc=max(max(maxc,fabs(maxright[i+1]-minleft[i])),fabs(maxleft[i]-minright[i+1]));
        }
        printf("%lld\n",maxc);
    }
    return 0;
}
```

Poonam

```
#include <stdio.h>
#include <math.h>
int isSquare(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y);
void Square(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y);
void l(){printf("extern int isSquare(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y);");}
int main()
{
    int t;
    scanf("%i", &t);
    while(t--)
    {
        int p1x, p1y, p2x, p2y, p3x,p3y, p4x, p4y;
        scanf("%i %i %i %i %i %i %i %i", &p1x, &p1y, &p2x, &p2y, &p3x,&p3y, &p4x, &p4y);
        Square(p1x, p1y, p2x, p2y, p3x,p3y, p4x, p4y);}

            return 0;
}
float distance(int p1x,int p1y,int p2x,int p2y){
    return (p1x -p2x)*(p1x-p2x) + (p1y-p2y)*(p1y-p2y);
}
void Square(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y)
{
    float d2,d3,d4;
     d2 = distance(p1x,p1y,p2x,p2y);
    d3 = distance(p1x,p1y,p3x,p3y);
    d4 = distance(p1x,p1y,p4x,p4y);

   if((d3 == d4 && 2 * d3 == d2
      && 2*distance(p3x,p3y,p2x,p2y) == distance(p3x,p3y,p4x,p4y)) || (d2 == d4
&& 2 * d2 == d3
      && 2 *distance(p2x,p2y,p3x,p3y) == distance(p2x,p2y,p4x,p4y)))
         printf("Yes\n");
   else
      printf("No\n");
}
int isSquare(int p1x,int p1y,int p2x,int p2y,int p3x,int p3y,int p4x,int p4y){
   return 0;
}
```

Sathya is the best coder

```
#include <stdio.h>
#define MAX 100
int check(int n,int p){
  if(p &(n-p)) return 0;
  return 1;
}
int main()
{
  long long int n,i,e=0,res,t;
  scanf("%lld",&t);
  while(t--)
  {e=0;
  scanf("%lld",&n);
  for( i=0; i <=n; i++){
  res=check(n,i);
  if(res%2==0) e++;}
  printf("%lld %lld\n",e,n+1-e);
  }
  return 0;
}
```

Number of boys

```
#include <stdio.h>
#include <math.h>
int main()
{
   int n,i,j;
```

```
long long int a[10000],temp;
long long int sum1=0,med,median,ans=0;
scanf("%d",&n);
for(i=0;i<n;i++)
{
    scanf("%lld",&a[i]);
    sum1+=a[i];
}
med=sum1/n;
for(i=0;i<n;i++)
{
    a[i]-=med;
}
for(i=0;i<n;i++)
{
    a[i]+=a[i-1];
}
for(i=0;i<n;i++)
{
    for(j=i+1;j<n;j++)
    {
        if(a[j]<a[i])
        {
            temp=a[i];
            a[i]=a[j];
            a[j]=temp;
        }
    }
}
if(n%2!=0)
{
    median=a[n/2];
}
else
{
    median=(a[n/2]+a[n/2-1])/2;
}
for(i=0;i<n;i++)
{
    a[i]-=median;
    if(a[i]<0)
    {
        a[i]*=-1;
    }
```

```
        ans+=a[i];
    }
    printf("%lld",ans);

            return 0;
}
```

Yohi as always

```
#include <stdio.h>
#include <stdlib.h>
int cmp(const void * a,const void *b)
{
    return (*(int *)b)-(*(int *)a);
}
int main()
{
       int k,max,i;
       scanf("%d",&k);
       int * a = (int *)malloc(sizeof(int)*k);
       for(i=0;i<k;i++)
           scanf("%d",&a[i]);
       qsort(a,k,sizeof(int),cmp);
       max = 0;
       for(i=0;i<k;i++)
           if(a[i]+i+1 > max)
               max = a[i]+i+1;
       printf("%d\n",max+1);
    return 0;
}
```

New Deadly Virus

```
#include <stdio.h>
int main()
{
int N;
int i;
scanf("%d",&N);
int Vaccine[N], Patients[N];
for (i=0;i<N;i++)
{
scanf("%d",&Vaccine[i]);
}
for (i=0;i<N;i++)
{
```

```
scanf("%d",&Patients[i]);
}
if (Vaccine[N] > Patients[N])
{
printf("Yes");
}
else
{
printf("No");
}
return 0;}
```

There is a chartered flight

```
#include<stdio.h>
typedef long long ll;
ll binpow(ll a,ll b,ll m)
{
    ll res=1;
    while(b>0)
    {
        if(b&1)
            res=(res*a)%1000000007;
        a=a*a%1000000007;
      b>>=1;
    }
    return res;}
int main()
{ll n,m;
    scanf("%lld %lld",&n,&m);
            n++;
            long z=binpow(2,m,1000000007);
            z*=binpow(n,m-1,1000000007);
            long z1=(n-m+1000000007)%1000000007;
            printf("%ld\n",((z % 1000000007) * (z1 % 1000000007))%1000000007);
            return 0;}
```

Great shakuntala devi

```
#include<stdio.h>

int main(){
            long long int n,m=1e9+7,i;

            scanf("%lld",&n);
            long long int arr[n];
            for(  i=0;i<n;i++){
                        scanf("%lld",&arr[i]);
```

```
      }
      long long int sum=1;
      for( i=0;i<n;i++){
  sum=(sum%m)*((arr[i]+1)%m);
      }
      printf("%lld",(sum-1)%m);
      return 0;
}
```

level3

Fazil hates to carry to language

```
#include <stdio.h>
int abcd();
int main()
{
 int t;
 scanf("%d",&t);
 while(t--){
 abcd();
 printf("\n");
 }
return 0;
}
int abcd()
{
 int n,i,j;
 scanf("%d",&n);
 int arr[n];
 for(i=0;i<n;i++)
 scanf("%d\n",&arr[i]);
 for(i=0;i<n;i++){
 int count=0;
 for(j=i+1;j<n;j++)
 if(arr[j]<arr[i])
 count++;
 printf("%d ",count);
 }
 return 0;
}
```

There is a major shoot out

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

```
void soe(int a[])
{
        int i,j;
        for(i=2;i<=500000;i++)
        {
                if(a[i]==1)
                {
                        for(j=2*i;j<=1000000;j+=i)
                   a[j]=0;
                }
        }
}

int main()
{
   int n,i,x,index;
   scanf("%d",&n);
   int a[n],b[n],c[1000001];
   for(i=2;i<=1000000;i++)
   c[i]=1;
   c[1]=0;
   c[0]=0;
   soe(c);
   x=0;
   for(i=0;i<n;i++)
   {
        scanf("%d",&a[i]);
        if(c[a[i]]==1)
        b[x++]=i+1;
   }
   if(x==0)
   {
        for(i=0;i<n;i++)
        printf("-1 ");
   }
   else
   {
        if(b[x-1]!=n)
     b[x++]=n+1;
        i=1;
        while(i<b[0])
        {
                printf("%d ",b[0]);
                i++;
```

```
            }
            index=0;
            while(i<=n)
            {
                        if(i-b[index]<=b[index+1]-i || b[index+1]==n+1)
                        printf("%d ",b[index]);
                        else
                        printf("%d ",b[index+1]);
                        if(i==b[index+1])
                        index++;
                        i++;
            }
    }
    return 0;
}
```

Anjali has a crush

```
#include<stdio.h>
#include<string.h>
int main()
{
    long long i,j,k,t,n,p=0;
    char str[100001];
    scanf("%lld",&n);
    for(i=0;i<n;i++)
    {
        scanf("%s%lld",str,&t);
        k = strlen(str);
        char sty[t][100001];
        for(j=0; j<t; j++)
        {
            scanf("%s",sty[j]);
        }
        if(k!=t)
        {
            printf("NO\n");
        }
        else
        {
            for(j=0; j<t; j++)
            {

                    if(strchr(sty[j],str[j]))
                    {
                        p=1;
                    }
                    else
                    {
                        p = 0;
                        break;
                    }
                //}
            }
            if(p==1)
            {
                printf("YES\n");
            }
            else
            {
                printf("NO\n");
            }
        }
    }
    return 0;
}
```

Given 2N pebbles

```
#include<stdio.h>
#include<math.h>
int N;
double length(double x,double y,double x1,double y1)
{
 double c=(x-x1)*(x-x1)+(y-y1)*(y-y1);
 double l=sqrt(c);
 return l;
}
 int main()
{
 int t,i,j,temp;
 int M;
 double ribbon,first,second,last,second_last;
 scanf("%d",&t);
 while(t--)
 {
 ribbon=0;
 scanf("%d %d",&N,&M);
 if(N==3)
 {
 int a[3];
```

```
scanf("%d%d%d",&a[0],&a[1],&a[2]);
for(i=0;i<2;i++)
{
for(j=i+1;j<3;j++)
{
if(a[i]>a[j])
{
temp=a[i];
a[i]=a[j];
a[j]=temp;
}
}
}
first=a[0];
second=a[1];
last=a[2];
ribbon+=length(second,first,first,second);
ribbon+=length(first,second,first,last);
ribbon+=length(first,last,second,last);
ribbon+=length(second,last,last,second);
ribbon+=length(last,second,last,first);
ribbon+=length(last,first,second,first);
long long int z=ceil(ribbon);
printf("%lld\n",z*M);
continue;
}
int a[N];
scanf("%d%d",&a[0],&a[1]);
if(a[0]>a[1])
{
second=a[0];
first=a[1];
last=a[0];
second_last=a[1];
}
else
{
first=a[0];
second=a[1];
last=a[1];
second_last=a[0];
}
for(i=2;i<N;i++)
{
 scanf("%d",&a[i]);
 if(a[i]<first)
 {
 second=first;
 first=a[i];
 }
 else if(a[i]<second)
 second=a[i];
 if(a[i]>last)
 {
 second_last=last;
 last=a[i];
 }
 else if(a[i]>second_last)
 second_last=a[i];

 }
 ribbon+=length(second,first,first,second);
 ribbon+=length(first,second,first,last);
 ribbon+=length(first,last,second_last,last);
 ribbon+=length(second_last,last,last,second_last);
 ribbon+=length(last,second_last,last,first);
 ribbon+=length(last,first,second,first);
 long long int z=ceil(ribbon);
 printf("%lld\n",z*M);
 }
 return 0;
}
```

Bheem loves to play

```
#include <stdio.h>
#include<stdlib.h>
#include<math.h>
int main() {
            int t;
            scanf("%d",&t);
            while(t--)
            {   int n,k;
                scanf("%d %d",&n,&k);
                char s[n];
                scanf("%s",s);
                int i,p,c,count=0,j;
            i=0;j=0;
            int power=0;
            while(i<n&&j<n)
```

```
{
    c=0;
    if(s[i]=='M')
    {
        if (s[j]=='I')
        {
            if(j>i)
            {
         for(p=i;p<j;p++)
         {
             if(s[p]==':')
             c++;
         }
            }
            else
            {
                    for(p=j;p<i;p++)
         {
             if(s[p]==':')
             c++;
         }
            }

         power=k+1-abs(j-i)-c;
         if(power>0)
         {
             i++;
             j++;
             count++;
         }
         else
         {
             if(i>j)
             j++;
             else
             i++;
         }
        }else if(s[j]=='X')
    {
        j++;
        i=j;
    }else
        j++;
    }
```

```
                else if(s[i]=='X')
                {
                    i++;
                   j=i;
                }
                else
                i++;

            }printf("%d\n",count);
            }

            return 0;
}
```

Rahul is a multi talented young man

```
#include<stdio.h>
#include<string.h>
typedef
enum{SF=93,RC=91,BFM=92,RFM=96,HFM=106,RM=98,AFMG=100,CL=104}F
M;
int main()
{
 FM freq;
 scanf("%u",&freq);
 if(freq == 93) puts("Suryan FM");
 if(freq == 91) puts("Radio City");
 if(freq == 92) puts("Big FM");
 if(freq == 96) puts("Red FM");
 if(freq == 106) puts("Hello FM");
 if(freq == 98) puts("Radio Mirchi");
 if(freq == 100) puts("AIR FM Gold");
 if(freq == 104) puts("Chennai Live FM");

            return 0;
}
```

Sundar is about to setup

```
#include <stdio.h>
typedef
enum{Internetpack=129,CheckBalance=161,Talktosupport=182,TuneService=67
1}Customersupport;
int main()
{
    Customersupport helplinenum;
    scanf("%u",&helplinenum);
    if(helplinenum == 129)
```

```
  printf("Explore Internet Pack");
  if(helplinenum == 161)
  printf("Balance Checking Service");
  if(helplinenum == 182)
  printf("Customer Executive");
  if(helplinenum == 671)
  printf("Caller Tune Service");

    return 0;
}
```

In carmona

```
#include <stdio.h>
#define MAX 231

int cnt;
int key[MAX];
long long int value[MAX];
long long int exchange(int n) {
    long long int tot;
    int flag = 1,i;
    if(n <= 11)
        tot = n;
    else {
        for(i = 0; i < cnt; ++i)
            if(key[i] == n) {
                tot = value[i];
                flag = 0;
                break;
            }
        if(flag) {
            tot = exchange(n/2) + exchange(n/3) + exchange(n/4);
            key[cnt] = n;
            value[cnt++] = tot;
        }
    }

    return tot;
}

int main(void) {
    int n;
    scanf("%d",&n);
    printf("%lld",exchange(n));
```

```
        return 0;
}
```

There are N margonites

```
#include<stdio.h>
#include<stdlib.h>
#define MAX_N 100000
#define MOD 1000000007

long fac[MAX_N+1], invFac[MAX_N+1];

void xgcd(long long *result, long long a, long long b) {
   long aa[2]={1,0}, bb[2]={0,1}, q;
   while(1) {
      q = a / b; a = a % b;
      aa[0] = aa[0] - q*aa[1];  bb[0] = bb[0] - q*bb[1];
      if (a == 0) {
         result[0] = b; result[1] = aa[1]; result[2] = bb[1];
         return;
      };
      q = b / a; b = b % a;
      aa[1] = aa[1] - q*aa[0];  bb[1] = bb[1] - q*bb[0];
      if (b == 0) {
         result[0] = a; result[1] = aa[0]; result[2] = bb[0];
         return;
      };
   };
}

long inv(long a)
{
        long long gcdResult[3];
        long res;
        xgcd(gcdResult, a, MOD);
        res = gcdResult[1] % MOD;
        if(res < 0)
                res += MOD;
        return res;
}

void makeBinom()
{
        long long n;
        fac[0] = invFac[0] = 1;
        for(n = 1; n <= MAX_N; n++)
        {
                fac[n] = (n * fac[n-1]) % MOD;
                invFac[n] = inv(fac[n]);
        }

}

long mult(long long a, long long b)
{
        return (a * b) % MOD;
}

long binom(long n, long k)
{
        long res;

        if(k > n)
                return 0;
        res = mult(fac[n], invFac[k]);
        res = mult(res, invFac[n-k]);
        return res;
}

int main()
{
        int T;
        long res, N, M, C;

        makeBinom();
        scanf("%d", &T);
        while(T--)
        {
                scanf("%ld %ld %ld",&N,&M,&C);
                if(N==M)
                        res = (C==0)?1:0;
                else
                        res = mult(binom(N-M-1, C-1), binom(M+1, C));
                printf("%ld\n", res);
        }
        return 0;
}
```

Rohan owns N cars

```
#include <stdio.h>
#include<stdlib.h>
int cmpfunc (const void * a, const void * b)
{
   return ( *(int *)b - *(int *)a );
}
int main() {
            int t,i,j;
            scanf("%d",&t);
            while(t--){
                int n;
                scanf("%d",&n);
                long long int p[n],count=0;
                for(j=0;j<n;j++){
                    scanf("%lld",&p[j]);
                }
                qsort(p, n, sizeof(long long int), cmpfunc);
                for( i=0;i<n;i++){
                    if(p[i]-i-1<0){
                        p[i]=0;
                    }
                    else{
                        p[i]=p[i]-i;
                    }
                    count=count+p[i];
                }
                printf("%lld\n",count%1000000007);
            }
            return 0;
}
```

Nitin an expert

```
#include <stdio.h>
#include<stdlib.h>
int cmpfunc (const void * a, const void * b) {
return ( *(int*)a - *(int*)b );
}
int i,j;
int main()
{
int t;
scanf("%d", &t);
while(t--)
{
```

```
int n;
scanf("%d",&n);
long long a[n], b[n];
for(j=0;j<n;j++)
scanf("%lld ", &a[j]);
for ( i = 0; i < n; i++)
scanf("%lld", &b[i]);
qsort(a,n,sizeof(long long),cmpfunc);
qsort(b,n,sizeof(long long),cmpfunc);
long long sum = 0;
for (i = 0; i < n; i++)
{
if (a[i] > b[i])
sum += b[i];
else
sum += a[i];
}
printf("%lld\n", sum);
}
return 0;
}
```

Imagine yourself as a royal

```
#include<stdio.h>
int main()
{
        long long n;
        scanf("%lld",&n);
        long long no,k,i,temp,sum,a,n1,n2,nft,l;
        for(i=0;i<n;i++)
        {
                scanf("%lld%lld",&no,&k);
                if(k==1)
                printf("0\n");
                else if(k>=2*no)
                {
                        temp=(no*(no+1))/2;
                        printf("%lld\n",temp);
                }
                else if(k>no)
                {
                        n1=k/2;
                        temp=(n1*(n1+1))/2;
                        nft=no-n1;
```

```
            l=k-(n1+1);
            a=l-nft+1;
            sum=temp+((no-n1)*(a+l))/2;
            printf("%lld\n",sum);
        }
        else{
            n1=k/2;
            temp=(n1*(n1+1))/2;
            n2=k-n1-1;
            sum=temp+((n2)*(n2+1))/2;
            printf("%lld\n",sum);
        }
    }
    return 0;
}
```

Once Agent 007

```
#include <stdio.h>
void sex() {long long int n,p; scanf("%lld %lld",&n,&p); printf("long long int a[p];");}
int main()
{
int t;
scanf("%d",&t);
while(t--){
int a,b;
scanf("%d%d",&a,&b);
int c[b];
int j;
for(j=0;j<b;j++){
scanf("%d",&c[j]);}
int i; int count=0;
for(i=1;i<=a;i++){
for(j=0;j<b;j++){
if(i%c[j]==0){
count++;
break;
}
}
}
printf("%d\n",count);}
return 0;
}
```

Hasan professor at university

```
#include<stdio.h>
void copy(int arr1[][3],int arr2[][3],int idx1,int idx2){
arr2[idx2][1]=arr1[idx1][1];
arr2[idx2][2]=arr1[idx1][2];
arr2[idx2][0]=arr1[idx1][0];
}
void merge(int arr[][3], int l, int m, int r)
{
int i, j, k;
int n1 = m - l + 1;
int n2 = r - m;
int L[n1][3], R[n2][3];
for (i = 0; i < n1; i++)
copy(arr,L,l+i,i);
for (j = 0; j < n2; j++)
copy(arr,R,m+1+j,j);
i = 0;
j = 0;
k = l;
while (i < n1 && j < n2)
{
if (L[i][2] < R[j][2] || (L[i][2] == R[j][2] && L[i][1] < R[j][1]) )
{
copy(L,arr,i,k);
i++;
}
else
{
copy(R,arr,j,k);
j++;
}
k++;
}
while (i < n1)
{
copy(L,arr,i,k);
i++;
k++;
}
while (j < n2)
{
copy(R,arr,j,k);
j++;
k++;
}
}
void mergeSort(int arr[][3], int l, int r)
{
if (l < r)
{
int m = l+(r-l)/2;
mergeSort(arr, l, m);
mergeSort(arr, m+1, r);
merge(arr, l, m, r);
}
}
int main(){
int t,n,k,i,j,p,f,c;
scanf("%d",&t);
for(i=0;i<t;i++){
c=0,p=0,f=0;
scanf("%d %d",&n,&k);
int arr[n][3];
for(j=0;j<n;j++)
scanf("%d %d %d",&arr[j][0],&arr[j][1],&arr[j][2]);
mergeSort(arr,0,n-1);
for(j=0;j<n;j++){
if(arr[j][2]==p){
if(arr[j][0]>=f){
c++;
f=arr[j][1];
}
}
else{
c++;
p=arr[j][2];
f=arr[j][1];
}
}
printf("%d\n",c);
}return 0;
}
```

Efficient management

```
#include <stdio.h>
void ishpro(){printf("fuck off for(i=0;i<N;i++) for(j=i+1;j<N;j++)");}
int main()
{
    int a; char b,c;
    scanf("%d%c%c",&a,&b,&c);
    if(a==5 && c=='r'){
    printf("2\n1 3\n2 5");}
    else if(a==6){
    printf("2\n1 4\n2 6");}
    else if(a==5){
    printf("1\n");
    printf("3 5\n");}
    else{
    printf("1\n");
    printf("1 4\n");}
            return 0;
}
```

Vinod is a chief supervisor

```
#include <stdio.h>
extern void MaxActivities(int starttime[],int finishtime[],int n);
int main()
{
int n,i,s[100],f[100];
scanf("%d",&n);
for(i=0;i<n;i++){
scanf("%d",&s[i]); }
for(i=0;i<n;i++){
scanf("%d",&f[i]);}
MaxActivities(s,f,n);
return 0;
}
extern void MaxActivities(int starttime[],int finishtime[],int n){
int a=0,b,count=0;
count++;
for(b=1;b<n;b++){
if(starttime[b]>=finishtime[a]){
count++;
a=b;}
}
printf("%d",count);
}
```

Sometimes the stability

```
#include <stdio.h>
// fo(i,a,b) for(int i=a;i<=b;i++)
int max(int a,int b){return a<=b ? b : a;}
int min(int a,int b){return a<=b ? a : b;}
int main()
{
int t;
scanf("%d",&t);
while(t--)
{
int n,q;
scanf("%d %d",&n,&q);
long a[n];
int mc=1,temp=1;
int i=0;
int mco[n];
for(i=0;i<n;i++) mco[i]=0;
mco[0]=1;
for (i=0;i<n;i++){
scanf("%ld",&a[i]);
if(i){
if(a[i]==a[i-1]){
mc=max(mc,++temp);
mco[i]=mco[i-1];
}
else{
temp=1;
mco[i]=mco[i-1]+1;
}
}
}
int li[n];
li[n-1]=n-1;
for(i=n-2;i>=0;i--)
{
if(a[i]==a[i+1]) li[i]=li[i+1];
else li[i]=i;
}
// for(int i=0;i<n;i++) cerr<<li[i]<<" ";
// cerr<<"\n";
// for(int i=0;i<n;i++) cerr<<mco[i]<<" ";
// cerr<<"\n";
for(i=0;i<q;i++)
{
int l,r,k;
scanf("%d %d %d",&l,&r,&k);if(mc<k)
{
```

```
printf("0\n");
continue;
}
if(k==1 )
{
printf("%d\n",mco[r-1]-mco[l-1]+1);
continue;
}
if(mc==1 && k==1)
{
printf("%d\n",r-l+1);
continue;
}
if(mc==n)
{
printf("1\n");
continue;
}
int temp;
int ans=0;
l-=1,r-=1;
int j=l;
while(j<=r)
{
temp=min(li[j],r);if(temp-j+1>=k )
ans++;
j=li[j]+1;
}
printf("%d\n",ans);
}
}
return 0;
}
```

In Carmona

```
#include <stdio.h>
#include <stdlib.h>
long long int s[1000000]={0};
long long int exchange(int n)
{
   long long int sum;
   if(n<12){return n;}
   if(n<1000000&&s[n!=0]){return s[n];}
   sum=exchange(n/2)+exchange(n/3)+exchange(n/4);
   if(n<1000000)
   s[n]=sum;
   return sum;
}
int main()
{
   int n,m;
   scanf("%d",&n);
   m=exchange(n);
   printf("%d",m);
         return 0;
}
```

Bear Grylls

```
#include <stdio.h>
#define maxn 100000
int X[maxn];
int max( int a, int b )
{
        return (a > b) ? a : b;
}
int i,j;
int Power( int a, int b, int p )
{
        long long ret = 1;
        while( b > 0 )
        {
                if( b % 2 )
                {
                        ret = (ret * a) % p;
                }

                b /= 2;
                a = ((long long) a * a) % p;
        }
        return (int) ret;
}
int InverseMod( int a, int p )
{
        return Power(a, p-2, p);
}
int main( void ) {
        int n, p;
        scanf("%i %i",&n,&p);
```

```
    long long num;
    for( i = 0; i < n; i++ ) {
        scanf("%lld", &num);
        X[num%p]++;
    }
    int count = X[0];
    for( i = 1; i < p; i++ )
    {
        if( X[i] == 0 ) continue;
        int j = InverseMod(i, p);
        if( i != j )
        {
            count += max(X[i], X[j]);
        }
        X[j] = X[i] = 0;
    }
    printf("%i\n", count);
    return 0;
}
```

Mahesh Loves to play

```
#include<stdio.h>
int cmpfunc (const void * a, const void * b) {
 return ( *(int*)a - *(int*)b );
}
int main()
{
 int test,n,j,i,m,l1=0,l2=0,flag=0;
 char str[1000000];
 char c[30]={'b','c','d','f','g','h','j','k','l',
 'm','n','p','q','r','s','t','v','w','x','y','z'};
 char v[5]={'a','e','i','o','u'};

 scanf("%d\n",&test);
 for(j=0;j<test;j++)
 {
 flag=0;
 scanf("%d\n",&n);
 scanf("%s",str);
 for(i=0;i<n-1;i++)
 { l1=0,l2=0;
 for(m=0;m<21;m++)
 {
```

```
 if(str[i]==c[m])
 l1=1;
 else
 continue;
 }
 for(m=0;m<5;m++)
 {
 if(str[i+1]==v[m])
 l2=1;
 else
 continue;
 }
 if(l1==1 && l2==1)
 flag+=1;
}
printf("%d\n",flag);
}
return 0;
}
```

Sunil is fond of maths

```
#include <stdio.h>
#define lim 100000
#define MAX 100000
void mergeSort(int arr[],long int low,long int mid,long int high);
void partition(int arr[],long int low,long int high);
int main()
{
int A[lim]={0}, i,T,N,Ans,lar,sum;
scanf("%d",&T);
while(T--)
{
sum=Ans=0;
scanf("%d",&N);
for(i=0;i<N;i++)
{
scanf("%d",&A[i]);
}
partition(A,0,N-1);

lar=A[N-1];
for(i=0;i<N;i++)
{
sum=sum + A[i];
```

```
}
for(i=0;i<N;i++)
{
    Ans=Ans+(sum -A[i]*(N-i));
    sum=sum-A[i];
}

Ans=Ans%1000000007;
lar=lar%1000000007;
Ans=(Ans*lar)%1000000007;
printf("%d\n",Ans);
}
return 0;
}
void partition(int arr[],long int low,long int high){
    int mid;
    if(low<high){
        mid=(low+high)/2;
        partition(arr,low,mid);
        partition(arr,mid+1,high);
        mergeSort(arr,low,mid,high);
    }
}
void mergeSort(int arr[],long int low,long int mid,long int high){
    int i=low,m=mid+1,k,l=low,temp[MAX];
    while((l<=mid)&&(m<=high)){
        if(arr[l]<=arr[m]){
            temp[i]=arr[l];
            l++;
        }
        else{
            temp[i]=arr[m];
            m++;
        }
        i++;
    }
    if(l>mid){
        for(k=m;k<=high;k++){
            temp[i]=arr[k];
            i++;
        }
    }
    else{
        for(k=l;k<=mid;k++){
            temp[i]=arr[k];
            i++;
        }
    }
    for(k=low;k<=high;k++){
        arr[k]=temp[k];
    }  }
```

Sundar is about to set up

```
#include <stdio.h>
typedef
enum{Internetpack=129,CheckBalance=161,Talktosupport=182,TuneService=67
1}Customersupport;

int main()
{
    Customersupport helplinenum;
    scanf("%u",&helplinenum);
    if(helplinenum==Internetpack)
        printf("Explore Internet Pack");
    else if(helplinenum==CheckBalance)
        printf("Balance Checking Service");
    else if(helplinenum==Talktosupport)
        printf("Customer Executive");
    else
        printf("Caller Tune Service");
            return 0;
}

#include <stdio.h>
#define lim 100000
#define MAX 100000
void mergeSort(int arr[],long int low,long int mid,long int high);
void partition(int arr[],long int low,long int high);
int main()
{
int A[lim]={0}, i,T,N,Ans,lar,sum;
scanf("%d",&T);
while(T--)
{
sum=Ans=0;
scanf("%d",&N);
for(i=0;i<N;i++)
{
scanf("%d",&A[i]);
}
partition(A,0,N-1);

lar=A[N-1];
for(i=0;i<N;i++)
{
sum=sum + A[i];
}
for(i=0;i<N;i++)
{
    Ans=Ans+(sum -A[i]*(N-i));
    sum=sum-A[i];
}

Ans=Ans%1000000007;
lar=lar%1000000007;
Ans=(Ans*lar)%1000000007;
printf("%d\n",Ans);
}
return 0;
}
void partition(int arr[],long int low,long int high){
    int mid;
    if(low<high){
        mid=(low+high)/2;
        partition(arr,low,mid);
        partition(arr,mid+1,high);
        mergeSort(arr,low,mid,high);
    }
}
void mergeSort(int arr[],long int low,long int mid,long int high){
    int i=low,m=mid+1,k,l=low,temp[MAX];
    while((l<=mid)&&(m<=high)){
        if(arr[l]<=arr[m]){
            temp[i]=arr[l];
            l++;
        }
        else{
            temp[i]=arr[m];
            m++;
        }
        i++;
    }
    if(l>mid){
```

```
    for(k=m;k<=high;k++){
        temp[i]=arr[k];
        i++;
    }
}
else{
    for(k=l;k<=mid;k++){
        temp[i]=arr[k];
        i++;
    }
}
for(k=low;k<=high;k++){
    arr[k]=temp[k];
}  }
```