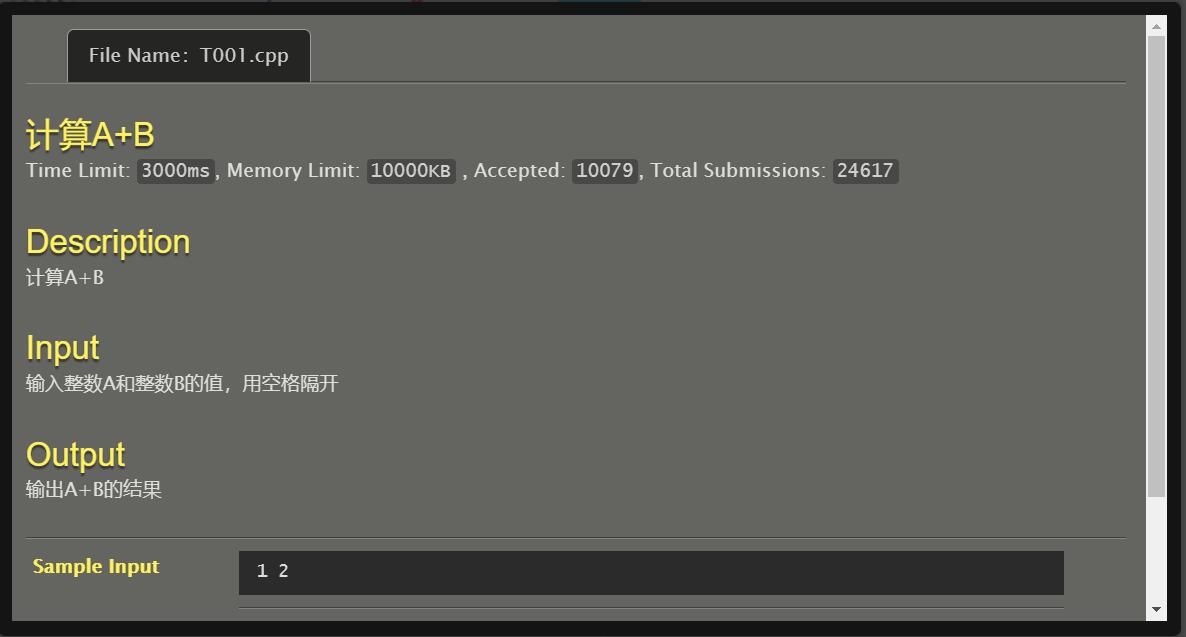
**西北工业大学NOJ题库集锦（By FZH）2018.6.20**

#include<stdio.h>

int main()

{

int a,b;

scanf("%d%d",&a,&b);

printf("%d\n",a+b);

return 0;

}



#include<stdio.h>

int main()

{

double r,h,l,s,sq,vq,vz,pi;

pi=3.141593;

scanf("%lf%lf",&r,&h);

l=2\*pi\*r;

s=pi\*r\*r;

sq=4\*pi\*r\*r;

vq=(4\*pi\*r\*r\*r)/3;

vz=s\*h;

printf("%.2lf\n%.2lf\n%.2lf\n%.2lf\n%.2lf\n",l,s,sq,vq,vz);

return 0;

}



#include<stdio.h>

int main()

{

double a,b,c,sum,ave;

scanf("%lf%lf%lf",&a,&b,&c);

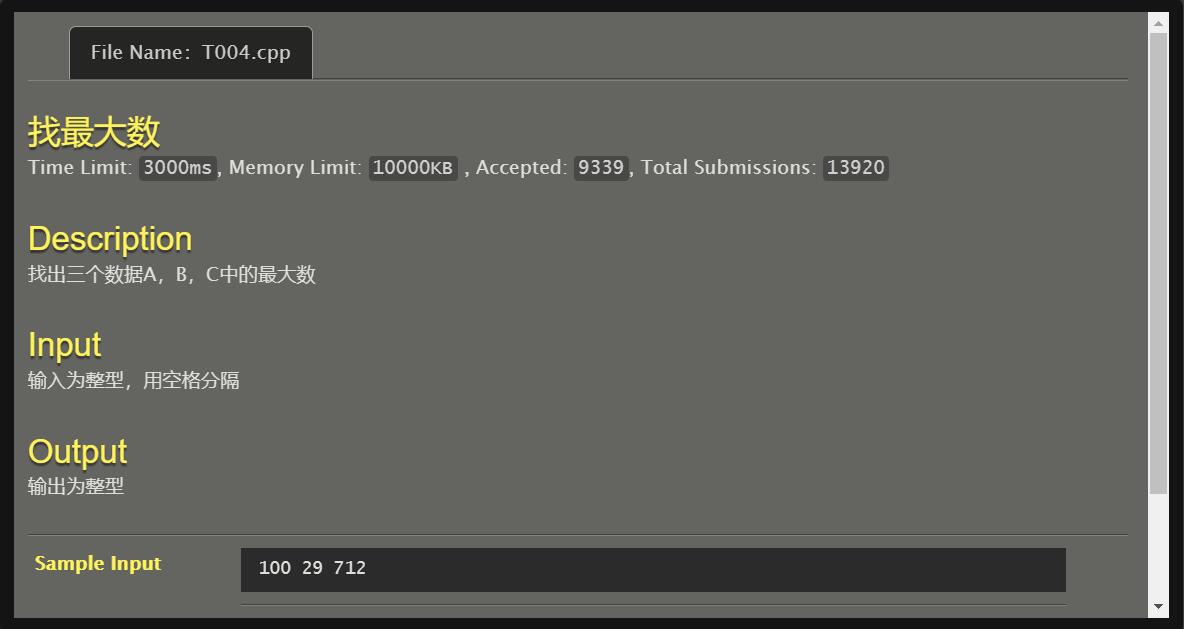
sum=a+b+c;

ave=sum/3;

printf("%.6lf\n%.6lf\n",sum,ave);

return 0;

}



#include<stdio.h>

int main()

{

int a,b,c,max;

scanf("%d%d%d",&a,&b,&c);

if (a>b) max=a;

else max=b;

if (c>max) max=c;

printf("%d",max);

return 0;

}



#include<stdio.h>

int main()

{

int i,n,m,q,p=0,sum=0,temp,ans;

scanf("%d",&n);

m=n;

ans=n;

while (m!=0)

{

m=m/10;

p++;

}

while (p>0)

{

p--;

q=n%10;

temp=1;

for (i=1;i<=p;i++) temp=temp\*10;

sum=sum+(q\*temp);

n=n/10;

}

if (sum==ans) printf("yes\n");

else printf("no\n");

return 0;

}



#include<stdio.h>

int main()

{

double i,ans;

scanf("%lf",&i);

if (i<=10) ans=i\*0.1;

else if ((i>10)&&(i<=20)) ans=(10\*0.1)+((i-10)\*0.075);

else if ((i>20)&&(i<=40)) ans=(10\*0.1)+(10\*0.075)+((i-20)\*0.05);

else if ((i>40)&&(i<=60)) ans=(10\*0.1)+(10\*0.075)+(20\*0.05)+((i-40)\*0.03);

else if ((i>60)&&(i<=100)) ans=(10\*0.1)+(10\*0.075)+(20\*0.05)+(20\*0.03)+((i-60)\*0.015);

else ans=(10\*0.1)+(10\*0.075)+(20\*0.05)+(20\*0.03)+(40\*0.015)+((i-100)\*0.01);

printf("%.6lf\n",ans);

return 0;

}



#include <stdio.h>

#include <math.h>

int main()

{

double s,ans;

scanf("%lf",&s);

if (s<=2) ans=7;

else if ((s>2)&&(s<=15)) ans=7+1.5\*(ceil(s)-2);

else ans=7+13\*1.5+2.1\*(ceil(s)-15);

printf("%.6lf",ans);

return 0;

}



#include<stdio.h>

int main()

{

int num[15];

int a,b,c,i,ans=0,flag=0;

scanf("%d-%d-%d",&a,&b,&c);

if ((a%4==0)&&(a%100!=0)) flag=1;

if ((a%100==0)&&(a%400==0)) flag=1;

num[1]=31;

num[3]=31;

num[4]=30;

num[5]=31;

num[6]=30;

num[7]=31;

num[8]=31;

num[9]=30;

num[10]=31;

num[11]=30;

num[12]=31;

if (flag==1) num[2]=29;

else num[2]=28;

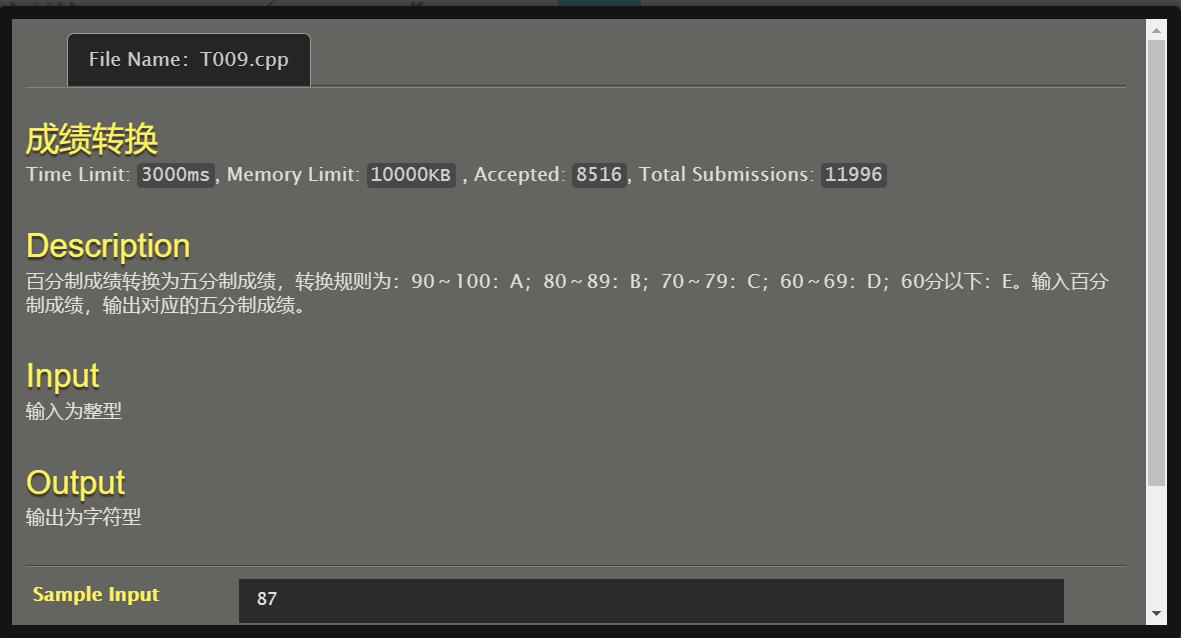
for (i=1;i<b;i++) ans=ans+num[i];

ans=ans+c;

printf("%d\n",ans);

return 0;

}



#include<stdio.h>

int main()

{

int n;

char grade;

scanf("%d",&n);

if (n>=90) grade='A';

else if (n>=80) grade='B';

else if (n>=70) grade='C';

else if (n>=60) grade='D';

else grade='E';

printf("%c\n",grade);

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

double x,y,d1,d2,d3,d4;

int h;

scanf("%lf,%lf",&x,&y);

d1=sqrt((x-2)\*(x-2)+(y-2)\*(y-2));

d2=sqrt((x+2)\*(x+2)+(y-2)\*(y-2));

d3=sqrt((x-2)\*(x-2)+(y+2)\*(y+2));

d4=sqrt((x+2)\*(x+2)+(y+2)\*(y+2));

if ((d1<=1)||(d2<=1)||(d3<=1)||(d4<=1)) h=10;

else h=0;

printf("%d\n",h);

return 0;

}



#include<stdio.h>

#include<math.h>

int prime(int k)

{

int j;

for (j=2;j<=trunc(sqrt(k));j++) if (k%j==0) return(1);

return(0);

}

int main()

{

int i,cnt=0,sum=0;

for (i=800;i>=500;i--)

{

if (prime(i)==0)

{

cnt++;

if (cnt%2!=0) sum+=i;

else sum-=i;

}

}

printf("%d %d",cnt,sum);

return 0;

}



#include<stdio.h>

int main()

{

int x,a,i,mi=1;

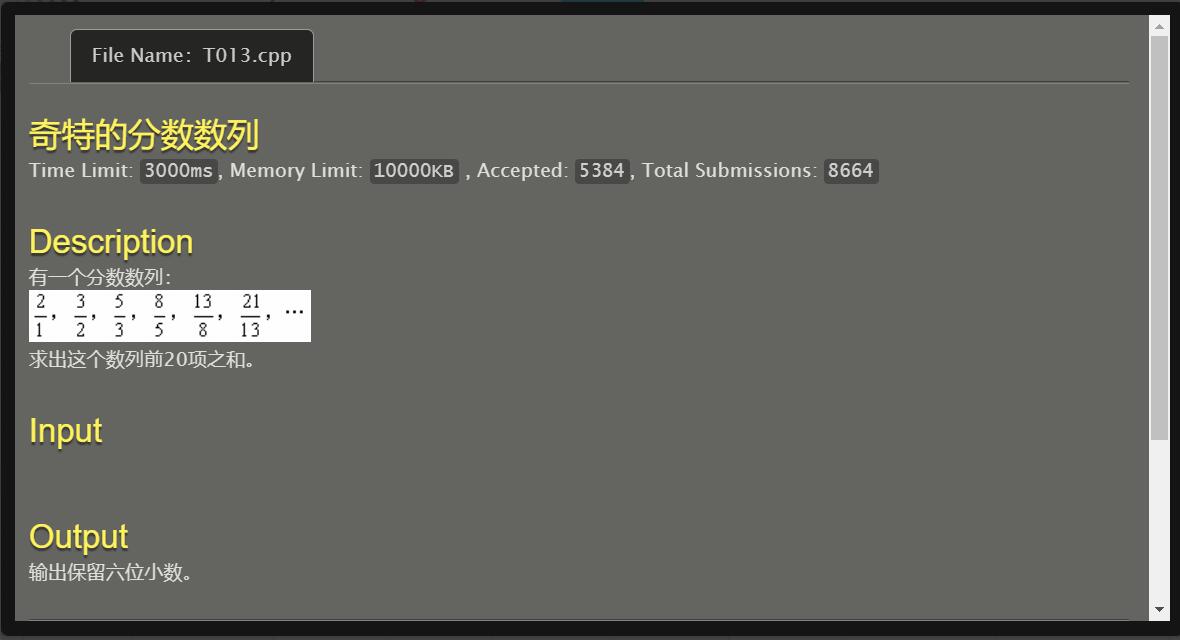
scanf("%d%d",&x,&a);

for (i=1;i<=a;i++) mi=(mi\*x)%1000;

printf("%d",mi);

return 0;

}



#include <stdio.h>

int main()

{

double a1=1,a2=2,a,b1=2,b2=3,b,sum=0;

int totol=3;

sum+=(b1/a1);

sum+=(b2/a2);

while (totol<=20)

{

a=a1+a2;

b=b1+b2;

sum+=(b/a);

a1=a2;a2=a;

b1=b2;b2=b;

totol++;

}

printf("%.6f",sum);

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

int i1,i2,i3,i5,i8,a1,a2,a3,a5,a8;

double sum,ans=0;

for (i8=0;i8<3;i8++)

for (i5=0;i5<=((20-i8\*8)/5);i5++)

for (i3=0;i3<=((20-i8\*8-i5\*5)/3);i3++)

for (i2=0;i2<=((20-i8\*8-i5\*5-i3\*3)/2);i2++)

{

i1=20-i8\*8-i5\*5-i3\*3-i2\*2;

sum=2000.0\*pow((1+0.0063\*12),i1)\*pow((1+2\*0.0066\*12),i2)

\*pow((1+3\*0.0069\*12),i3)\*pow((1+5\*0.0075\*12),i5)\*pow((1+8\*0.0084\*12),i8);

if (sum>ans)

{

ans=sum;

a1=i1;a2=i2;a3=i3;

a5=i5;a8=i8;

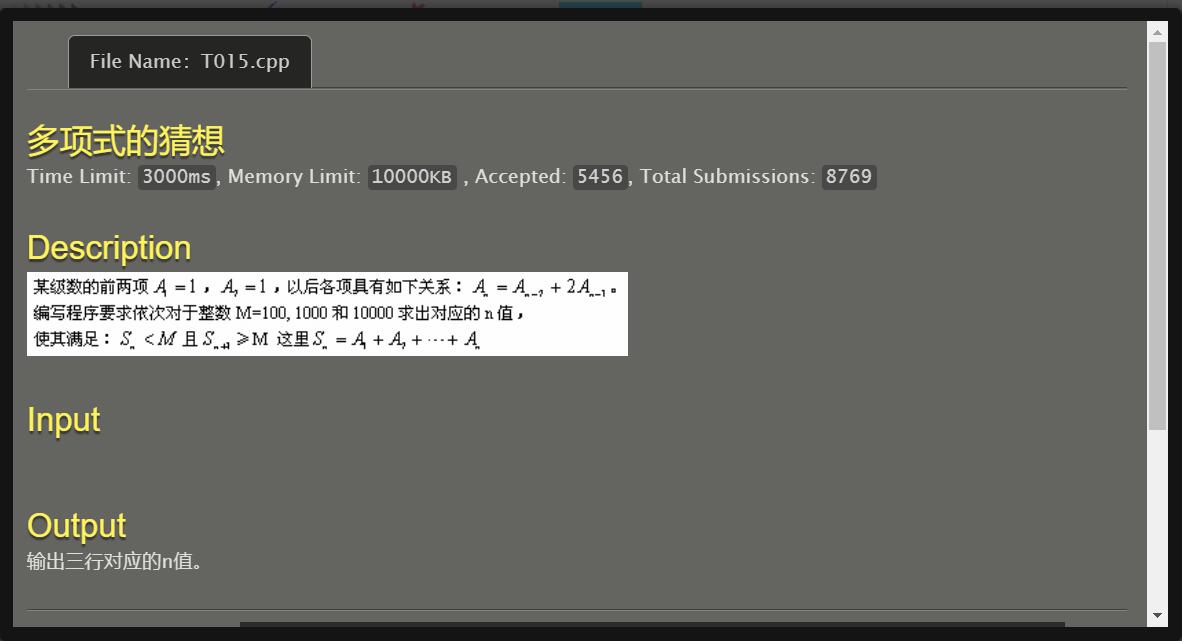
}

}

printf("%d %d %d %d %d\n%.2lf",a8,a5,a3,a2,a1,ans);

return 0;

}



#include<stdio.h>

int main()

{

int a1=1,a2=1,a,sum=2,cnt=2;

while (sum<10000)

{

a=2\*a2+a1;

a1=a2;

a2=a;

if (((sum<100)&&(sum+a>=100))||((sum<1000)&&(sum+a>=1000))||((sum<10000)&&(sum+a>=10000))) printf("%d\n",cnt);

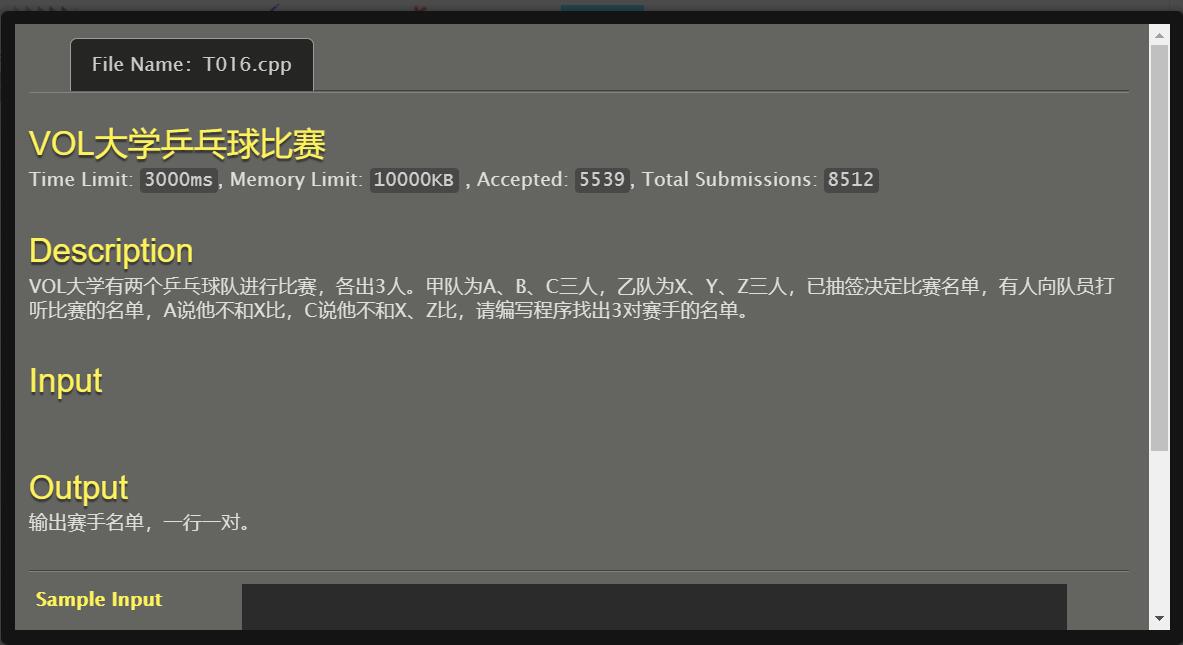
sum+=a;

cnt++;

}

return 0;

}



#include <stdio.h>

int main()

{

int a,b,c;

a=90;

b=88;

c=89;

printf("A=%c\nB=%c\nC=%c\n",a,b,c);

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

double a,x1,x2=0,temp,ans;

scanf("%lf",&a);

x1=a;

while (fabs(x1-x2)>=0.00001)

{

temp=x1;

x2=0.5\*(x1+(a/x1));

x1=x2;

x2=temp;

}

ans=x1;

printf("%.5f",ans);

return 0;

}



#include <stdio.h>

int main()

{

int n,s3,s2,a1,an,i;

scanf("%d",&n);

s3=n\*n\*n;

s2=n\*n;

if (n%2!=0) a1=s2-2\*(n/2);

else a1=s2-2\*(n/2)+1;

an=a1+2\*(n-1);

printf("%d\*%d\*%d=%d=",n,n,n,s3);

for (i=0;i<n-1;i++) printf("%d+",a1+2\*i);

printf("%d\n",an);

return 0;

}



#include<stdio.h>

int main()

{

int n,j,i,k,jsum;

double sum=0,t1,t2,t3;

scanf("%d",&n);

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

{

jsum=1;

if (j%2==0) i=1;

else i=-1;

for (k=1;k<=j;k++) jsum=jsum\*2;

t1=(-1)\*i\*jsum;

t2=jsum+i;

t3=2\*jsum-i;

sum=sum+(t1/(t2\*t3));

}

printf("%.6f",sum);

return 0;

}



/\*二分法求根\*/

#include <stdio.h>

int main()

{

double l,r,x,ls,xs;

scanf("%lf %lf",&l,&r);

while (((2\*l\*l\*l-4\*l\*l+3\*l-6)!=0)&&((2\*r\*r\*r-4\*r\*r+3\*r-6)!=0))

{

x=(l+r)/2;

xs=2\*x\*x\*x-4\*x\*x+3\*x-6;

ls=2\*l\*l\*l-4\*l\*l+3\*l-6;

if ((ls\*xs)<0) r=x;

else l=x;

}

if ((2\*l\*l\*l-4\*l\*l+3\*l-6)==0) printf("%.2f",l);

else printf("%.2f",r);

return 0;

}



#include <stdio.h>

int main()

{

int m,n,g=0,s=0,nmin=1,nmax,i,j,k,a,flag;

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

for (i=1;i<n;i++) nmin=nmin\*10;

nmax=nmin\*10-1;

for (j=nmin;j<=nmax;j++)

{

k=j;

while (k!=0)

{

flag=0;

a=k%10;

k=k/10;

if (a==m)

{

flag=1;

break;

}

}

if ((flag==1)&&((j%m)!=0))

{

g++;

s+=j;

}

}

printf("%d %d",g,s);

return 0;

}



#include <stdio.h>

int main()

{

int a,b,c,t,i;

scanf("%d%d%d",&a,&b,&c);

printf("%d.",a/b);

a=a%b;

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

{

printf("%d",(a\*10)/b);

a=(a\*10)%b;

}

t=((a\*100)/b)%10;

if (t<=4) printf("%d",(a\*10)/b);

else printf("%d",((a\*10)/b)+1);

return 0;

}



#include <stdio.h>

int main()

{

int a,b,c,m,n,k,x1,x2,x3,ans=101;

scanf("%d%d%d",&a,&b,&c);

for (m=1;m<33;m++)

{

x1=3\*m+a;

for (n=1;n<20;n++)

{

x2=5\*n+b;

for (k=1;k<14;k++)

{

x3=7\*k+c;

if ((x1==x2)&&(x2==x3)&&(x1<ans))

ans=x1;

}

}

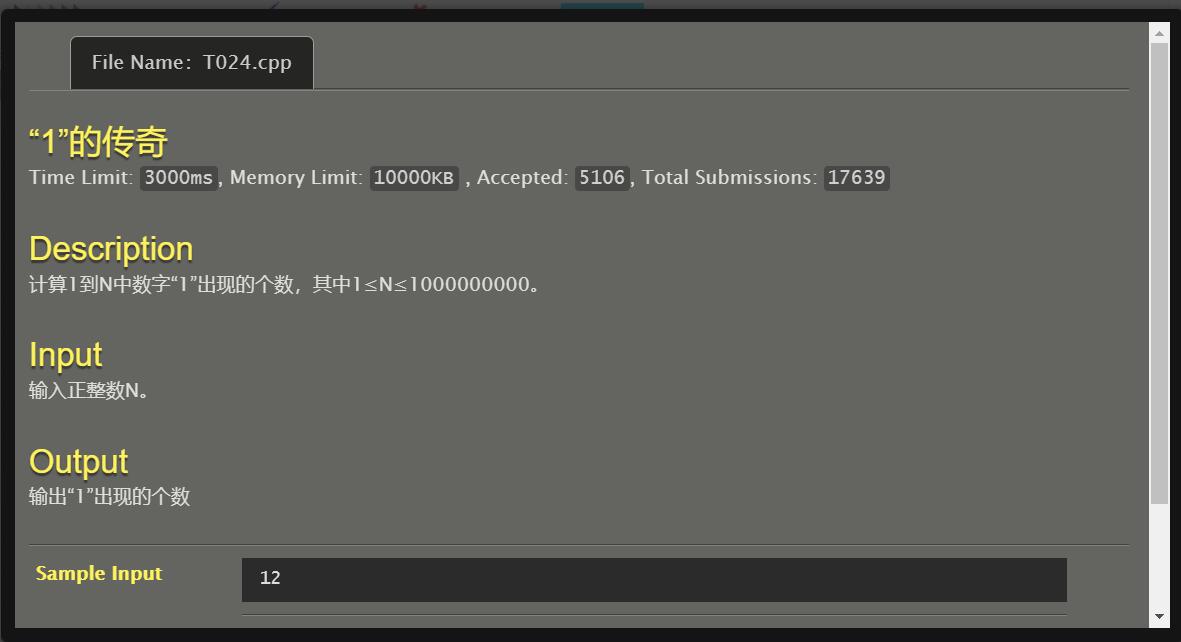
}

if (ans!=101) printf("%d",ans);

else printf("-1");

return 0;

}



#include <stdio.h>

int main()

{

int i,n,t1,t2,c=1,w=0,p,x[20],ans=0;

scanf("%d",&n);

t1=n;t2=n;

while (t1!=0)

{

t1=t1/10;

w++;

}

x[0]=0;

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

{

x[i]=10\*(x[i-1])+c;

c=c\*10;

}

while (t2!=0)

{

w--;

p=t2/c;

if (p==1) ans=ans+(t2-c+1)+(p\*x[w]);

else ans=ans+c+(p\*x[w]);

t2=t2%c;

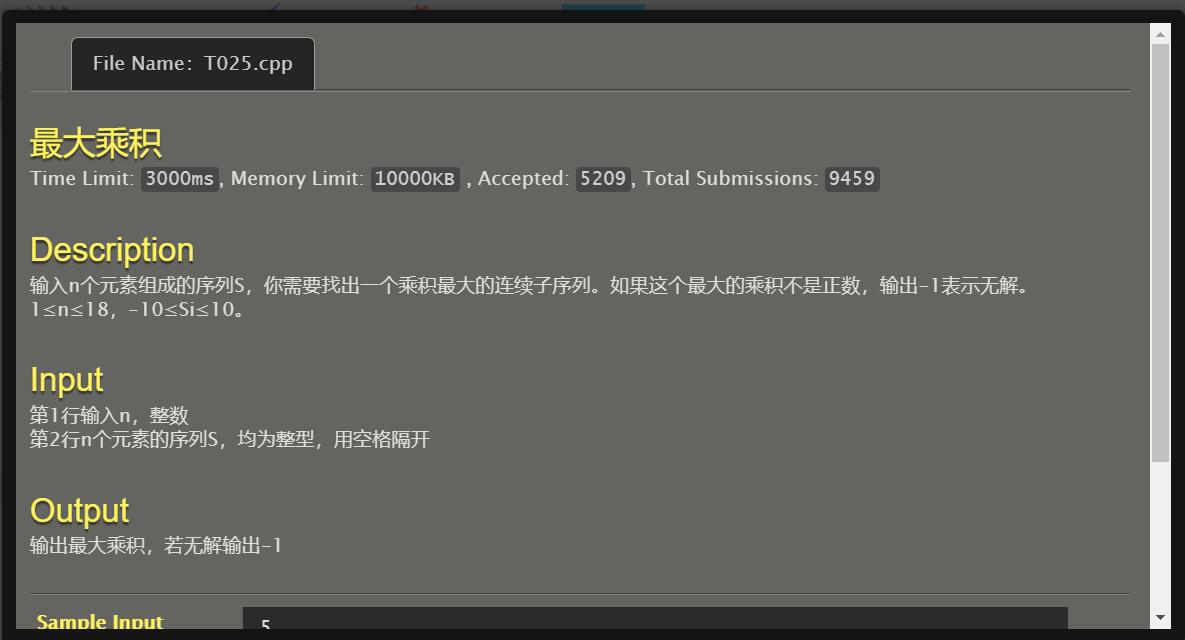
c=c/10;

}

printf("%d",ans);

return 0;

}



#include <stdio.h>

int main()

{

int n,k,p,i,j,s[20],ans=0,cur;

scanf("%d",&n);

for (k=1;k<=n;k++) scanf("%d",&s[k]);

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

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

{

cur=1;

for (p=i;p<=j;p++) cur=cur\*s[p];

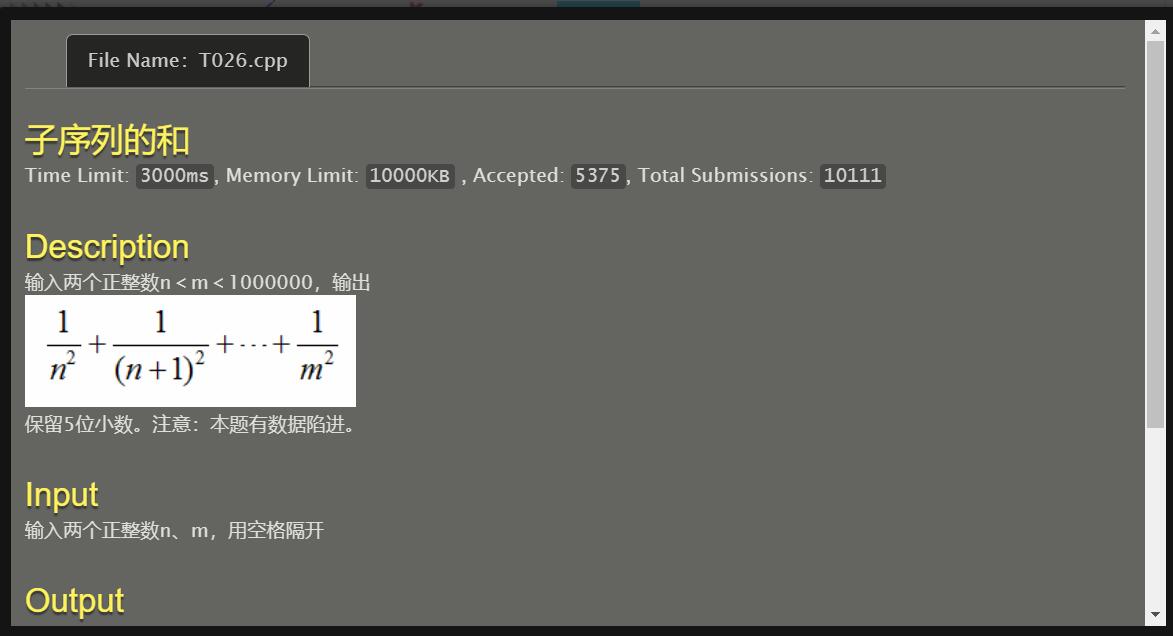
if (cur>ans) ans=cur;

}

printf("%d",ans);

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

int n,m,i;

double x,ans=0;

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

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

{

x=pow(i,2.0);

ans+=(1/x);

}

printf("%.5lf",ans);

return 0;

}



#include<stdio.h>

#include<math.h>

int solve(int k)

{

int cnt;

if (k==1||k==2) cnt=0;

else if (k==3) cnt=1;

else if (k==4) cnt=3;

else cnt=(2\*solve(k-1)+pow(2,k-4)-solve(k-4));

return cnt;

}

int main()

{

int n;

scanf("%d",&n);

printf("%d",solve(n));

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

int k,x,y;

double p,q,w;

scanf("%d",&k);

for (y=k+1;y<=2\*k;y++)

{

x=(k\*y)/(y-k);

p=1.0/k;

q=1.0/x;

w=1.0/y;

if ((fabs(q+w-p)<=1e-5)&&(x>=y))

{

printf("1/%d=1/%d+1/%d\n",k,x,y);

}

}

return 0;

}



#include <stdio.h>

int main()

{

int a,b,n=1,cnt=1;

double sumd=1,sum=1;

scanf("%d%d",&a,&b);

while (sum<b)

{

if (((sum>a)&&(cnt==1))||((sum+(1.0/(sumd+(1.0/(n+1))))>=b)))

{

printf("%d ",n);

cnt++;

}

n++;

sumd+=(1.0/n);

sum+=(1.0/sumd);

}

return 0;

}



#include<stdio.h>

#include<math.h>

int prime(int k)

{

int c,flag=0;

for (c=2;c<sqrt(k);c++) if (k%c==0)

{

flag=1;

break;

}

return(flag);

}

int main()

{

int n,j=0,cnt=0,l=0,r=99;

scanf("%d",&n);

while (cnt<n)

{

for (j=l;j<=r;j++)

{

if (j%2==0) continue;

if (j%3==0) continue;

if (j%5==0) continue;

if (j%7==0) continue;

if (j%11==0) continue;

if (j%13==0) continue;

if (j%17==0) continue;

if (j%19==0) continue;

if (j%23==0) continue;

if (prime(j)==0) break;

}

if (j==(r+1))

{

cnt++;

if (cnt==n) printf("%d %d\n",l,r);

}

l+=100;

r+=100;

}

return 0;

}



#include<stdio.h>

#include<math.h>

int fceil(double k)

{

int ans;

if (k>0) ans=k+1;

else if (fabs(k)<=1e-6) ans=0;

else ans=k;

return(ans);

}

int main()

{

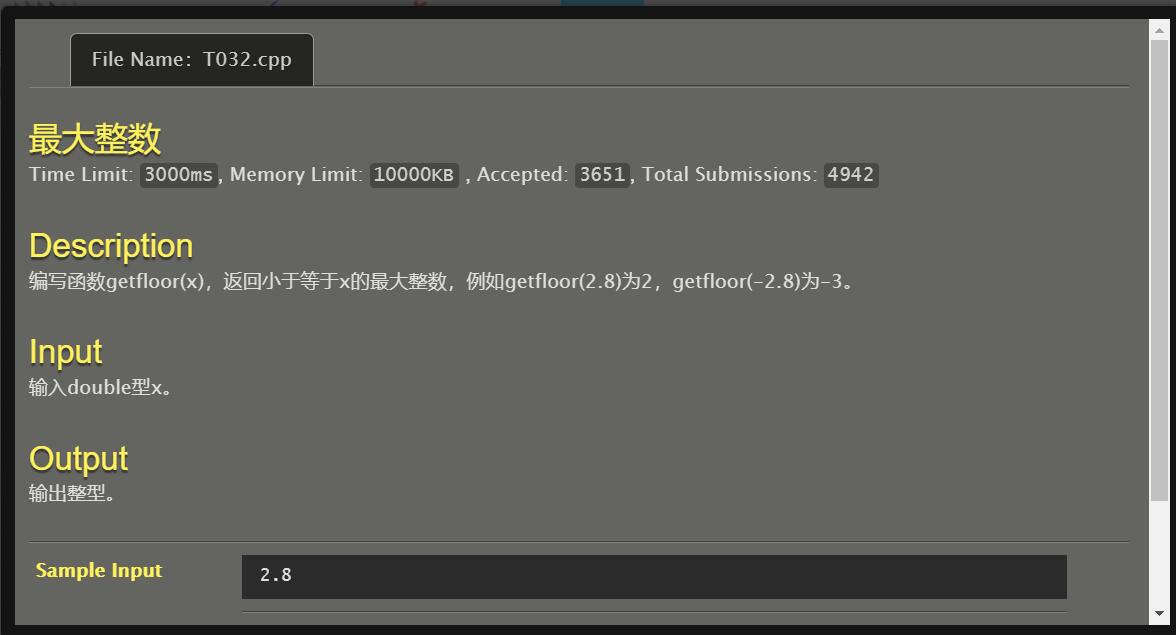
double x;

scanf("%lf",&x);

printf("%d",fceil(x));

return 0;

}



#include<stdio.h>

#include<math.h>

int getfloor(double k)

{

int ans;

if (k>0) ans=k;

else if (fabs(k)<=1e-6) ans=0;

else ans=k-1;

return(ans);

}

int main()

{

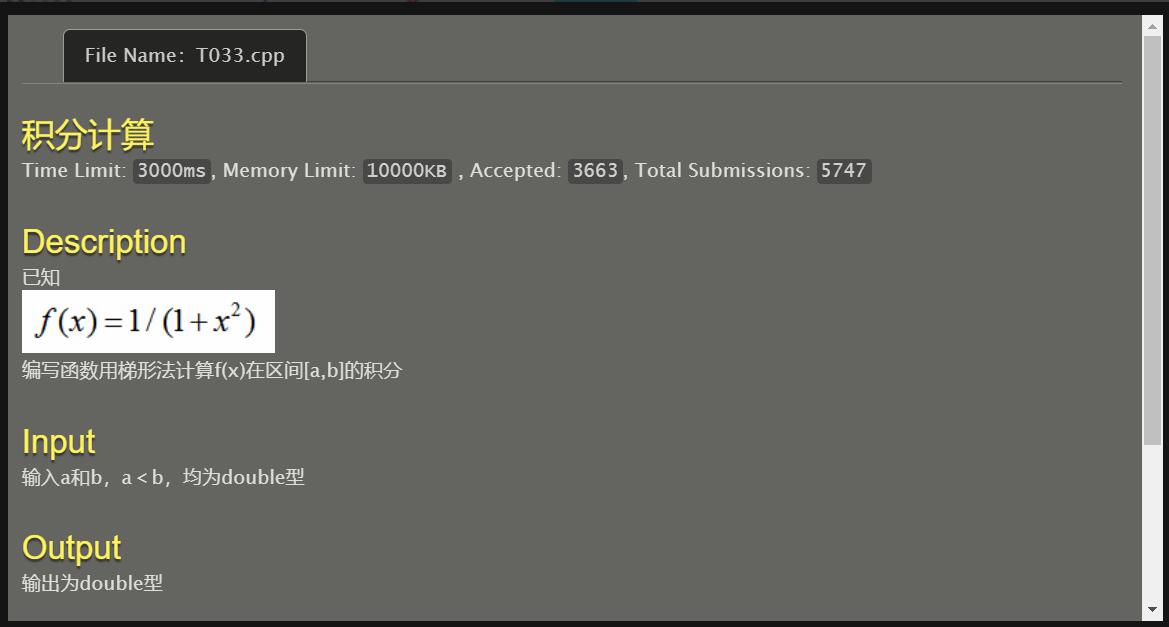
double x;

scanf("%lf",&x);

printf("%d",getfloor(x));

return 0;

}



#include <stdio.h>

double return\_result(double k)

{

return((1.0/(1+(k\*k))));

}

int main()

{

double a,b,ans=0,gap=1e-6,s;

scanf("%lf%lf",&a,&b);

s=a;

while (s<=b)

{

ans+=(gap\*return\_result(s));

s+=gap;

}

printf("%f\n",ans);

return 0;

}



#include <stdio.h>

int main()

{

int n,i,tot=4,temp1,temp2,w[5];

scanf("%d",&n);

while (n!=0)

{

w[tot]=(n%10);

n=n/10;

tot--;

}

for (i=1;i<=4;i++) w[i]=((w[i]+5)%10);

temp1=w[1];

w[1]=w[4];

w[4]=temp1;

temp2=w[2];

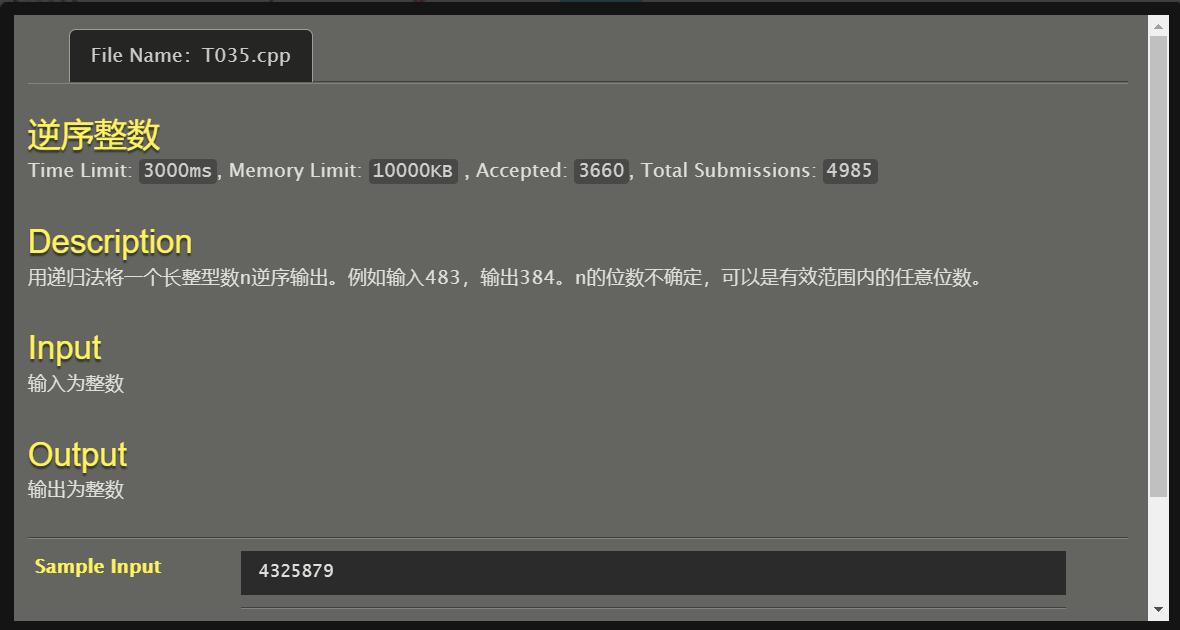
w[2]=w[3];

w[3]=temp2;

for (i=1;i<=4;i++) printf("%d",w[i]);

return 0;

}



#include <stdio.h>

void print\_reverse(int k)

{

if (k==0) return;

printf("%d",(k%10));

print\_reverse(k/10);

}

int main()

{

int n;

scanf("%d",&n);

print\_reverse(n);

return 0;

}



#include <stdio.h>

int salary(int k)

{

int c;

c=0;

if (k==0) return(c);

else if ((k/100)>0) c+=((k/100)+salary(k%100));

else if ((k/50)>0) c+=((k/50)+salary(k%50));

else if ((k/10)>0) c+=((k/10)+salary(k%10));

else if ((k/5)>0) c+=((k/5)+salary(k%5));

else if ((k/2)>0) c+=((k/2)+salary(k%2));

else c+=((k/1)+salary(k%1));

return(c);

}

int main()

{

int n,x,i,ans=0;

scanf("%d",&n);

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

{

scanf("%d",&x);

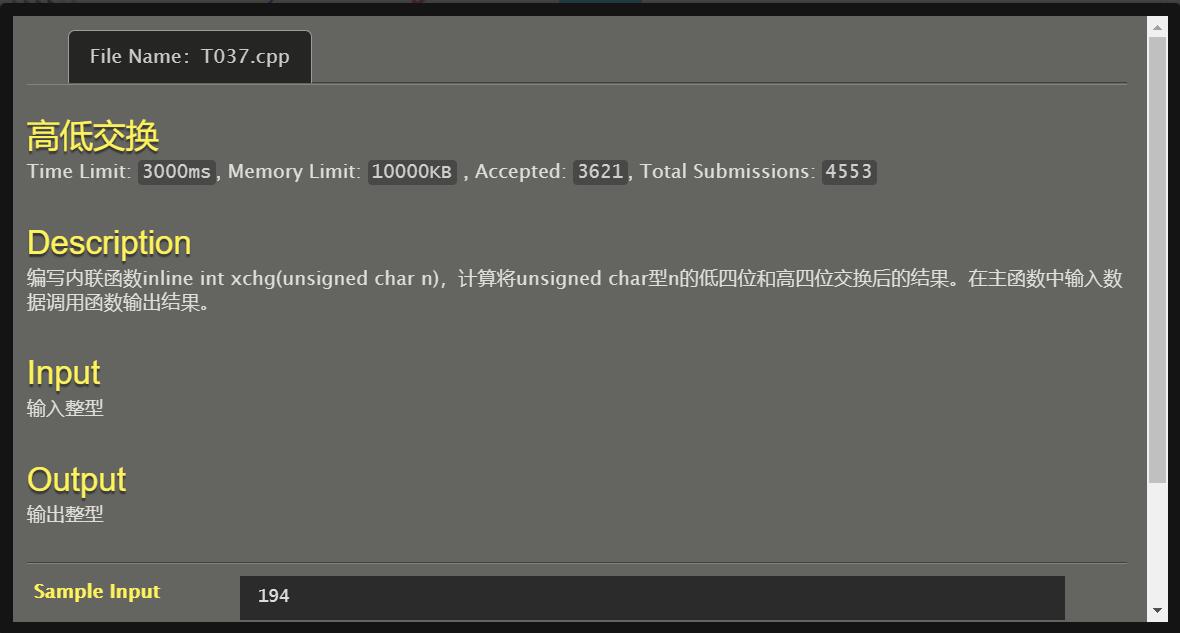
ans+=salary(x);

}

printf("%d",ans);

return 0;

}



#include <stdio.h>

inline int xchg(unsigned char n)

{

n=n>>4|n<<4;

return(n);

}

int main()

{

int n;

scanf("%d",&n);

printf("%d\n",xchg(n));

return 0;

}



#include <stdio.h>

int main()

{

int ah,am,as,bh,bm,bs,ch,cm,cs,sum=0;

scanf("%d%d%d%d%d%d",&ah,&am,&as,&bh,&bm,&bs);

sum+=(ah\*3600+am\*60+as+bh\*3600+bm\*60+bs);

ch=sum/3600;

sum=sum%3600;

cm=sum/60;

cs=sum%60;

printf("%d %d %d\n",ch,cm,cs);

return 0;

}



#include<stdio.h>

#include<math.h>

int main()

{

int n,i;

double x,fangcha,sum1=0,sum2=0,pj;

scanf("%d",&n);

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

{

scanf("%lf",&x);

sum1+=pow(x,2);

sum2+=x;

}

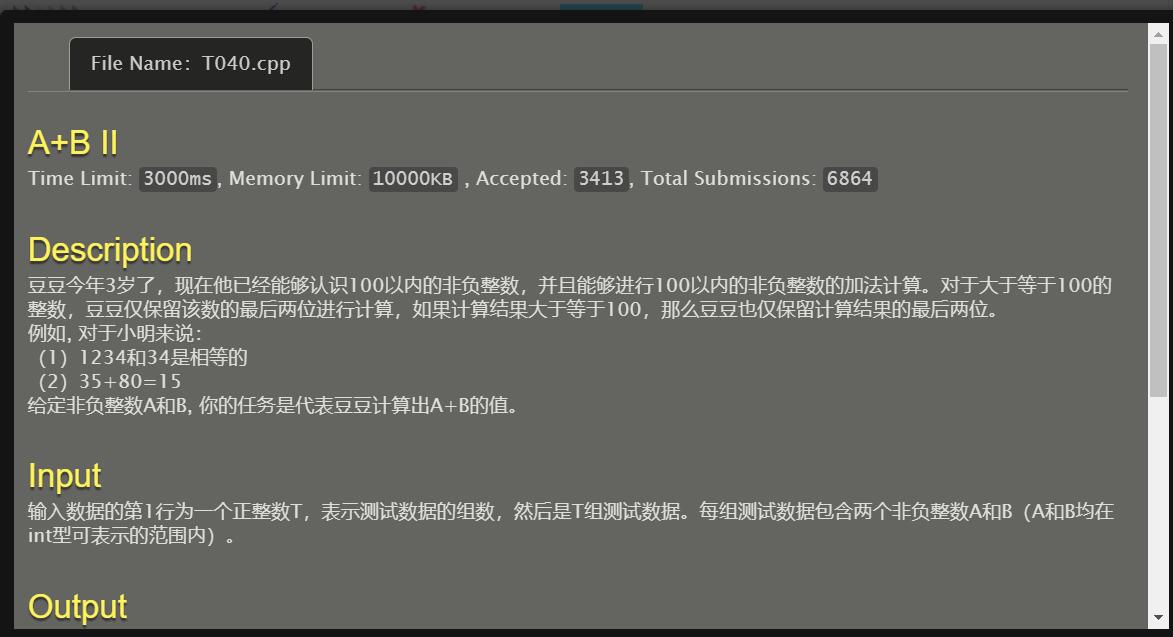
pj=sum2/(n\*1.0);

fangcha=((sum1)-(2\*(sum2)\*pj)+(n\*(pj)\*(pj)));

printf("%.6f\n",fangcha);

return 0;

}



#include <stdio.h>

int plus(int a,int b)

{

int c;

c=((a%100)+(b%100))%100;

return(c);

}

int main()

{

int t,i,a,b;

scanf("%d",&t);

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

{

scanf("%d%d",&a,&b);

printf("%d\n",plus(a,b));

}

return 0;

}



#include <stdio.h>

void QuickSort(int A[105],int l,int r)

{

int temp,k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (A[++i]>A[k]));

while ((j>=l) && (A[--j]<A[k]));

if (i>=j) break;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

temp=A[k];

A[k]=A[j];

A[j]=temp;

QuickSort(A,l,j-1);

QuickSort(A,j+1,r);

}

}

int main()

{

int n,s,m,i,a[105];

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d%d",&s,&m);

QuickSort(a,s,m);

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

printf("\n");

return 0;

}



#include <stdio.h>

int main()

{

int a[105],b[105],c[105],n,i;

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

for (i=0;i<n;i++) scanf("%d",&b[i]);

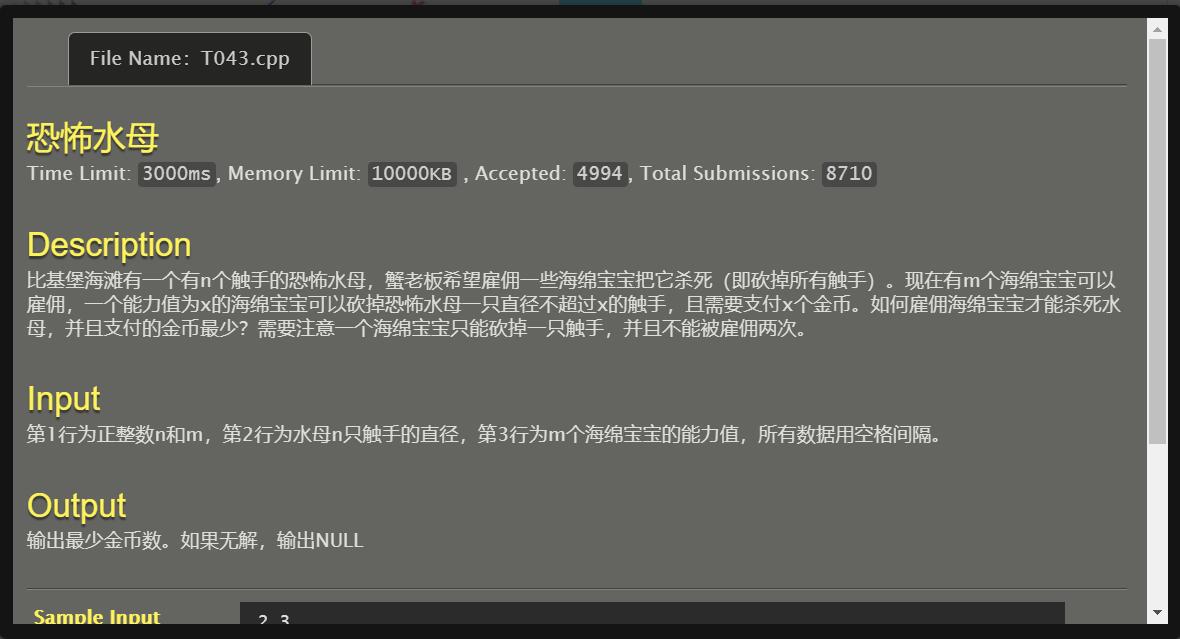
for (i=0;i<n;i++) c[i]=a[i]+b[i];

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

printf("\n");

return 0;

}



#include <stdio.h>

void QuickSort(int A[10005],int l,int r)

{

int temp,k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (A[++i]<A[k]));

while ((j>=l) && (A[--j]>A[k]));

if (i>=j) break;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

temp=A[k];

A[k]=A[j];

A[j]=temp;

QuickSort(A,l,j-1);

QuickSort(A,j+1,r);

}

}

int main()

{

int n,m,sum=0,i,j,k,flag=0,d[10005],x[10005];

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

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

for (i=0;i<m;i++) scanf("%d",&x[i]);

QuickSort(d,0,n-1);

QuickSort(x,0,m-1);

i=0;k=0;

while (i<=n-1)

{

j=k;

if (j>=m)

{

flag=1;

break;

}

while (x[j]<d[i]) j++;

sum+=x[j];

k=j+1;

i++;

}

if (flag==0) printf("%d",sum);

else printf("NULL");

return 0;

}



#include <stdio.h>

void BubbleSort(int A[],int s,int m)

{

int i,j,temp;

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

{

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

if (A[j]<A[j+1])

{

temp=A[j];

A[j]=A[j+1];

A[j+1]=temp;

}

}

}

int main()

{

int n,s,m,i,a[105];

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d%d",&s,&m);

BubbleSort(a,s,m);

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

printf("\n");

return 0;

}



#include <stdio.h>

void SelectionSort(int A[],int s,int m)

{

int i,j,min,temp;

for (i=s;i<(s+m-1);i++)

{

min=i;

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

{

if (A[min]<A[j])

{

min=j;

}

}

if (min!=i)

{

temp=A[min];

A[min]=A[i];

A[i]=temp;

}

}

}

int main()

{

int n,s,m,i,a[105];

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d%d",&s,&m);

SelectionSort(a,s,m);

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

printf("\n");

return 0;

}



#include <stdio.h>

int main()

{

int a[105][105],n,i,j;

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 (j<i) printf(" ");

else printf("%d ",a[i][j]);

}

printf("\n");

}

return 0;

}



#include <stdio.h>

int main()

{

int a[105],b[105],n,i;

scanf("%d",&n);

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

{

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

b[i]=a[i];

}

for (i=0;i<n;i++) if ((i%2)!=0) printf("%d ",b[i]);

printf("\n");

return 0;

}



#include <stdio.h>

double avg(int A[],int s,int e)

{

double c,sum=0;

int j;

for (j=s;j<=e;j++) sum+=A[j];

c=sum/((e-s+1)\*1.0);

return (c);

}

int main()

{

int n,i,s,e,a[105];

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d%d",&s,&e);

printf("%f",avg(a,s,e));

return 0;

}



#include <stdio.h>

int BinarySearch(int A[105],int n,int find)

{

int low,upper,mid;

low=0;

upper=n-1;

while (low<=upper)

{

mid=low+(upper-low)/2;

if (A[mid]>find) upper=mid-1;

else if (A[mid]<find) low=mid+1;

else return (mid);

}

return (-1);

}

int main()

{

int n,i,a[105],m,ans;

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d",&m);

ans=BinarySearch(a,n,m);

if (ans!=-1) printf("%d\n",ans);

else printf("null\n");

return 0;

}



#include <stdio.h>

void InsertionSort(int A[],int s,int m)

{

int i,j,temp,cnt=0;

while (cnt<=m)

{

i=s;

temp=A[i];

j=i+1;

while ((j<(s+m))&&(temp<A[j]))

{

A[j-1]=A[j];

j++;

}

A[j-1]=temp;

cnt++;

}

}

int main()

{

int n,s,m,i,a[105];

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d",&a[i]);

scanf("%d%d",&s,&m);

InsertionSort(a,s,m);

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

printf("\n");

return 0;

}



#include<stdio.h>

#include<string.h>

int number\_flag(char c)

{

int flag=0;

if ((c>='1')&&(c<='9')) flag=1;

return (flag);

}

int main()

{

char str[1005],s[1005],e[1005];

int cnt=0,i=0,j,k,a,b,len;

gets(str);

len=strlen(str);

while (i<len)

{

if (number\_flag(str[i]))

{

cnt++;

s[cnt]=i;

while (number\_flag(str[i])==1) i++;

i=i-1;

e[cnt]=i;

}

i++;

}

printf("%d\n",cnt);

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

{

a=s[j];

b=e[j];

for (k=a;k<=b;k++) printf("%c",str[k]);

printf(" ");

}

return 0;

}



#include<stdio.h>

#include<string.h>

void Left(char src[],int n,char dest[])

{

int i;

strncpy(dest,src,n);

for (i=0;i<n;i++) printf("%c",dest[i]);

printf("\n");

}

void Right(char src[],int n,char dest[])

{

int i,len,s;

len=strlen(src);

s=len-n;

for (i=s;i<len;i++) printf("%c",src[i]);

printf("\n");

}

void Mid(char src[],int loc,int n,char dest[])

{

int i,e;

e=loc+n;

for (i=loc;i<e;i++) printf("%c",src[i]);

}

int main()

{

char s[1005],dest[1005];

int n,loc;

gets(s);

scanf("%d%d",&n,&loc);

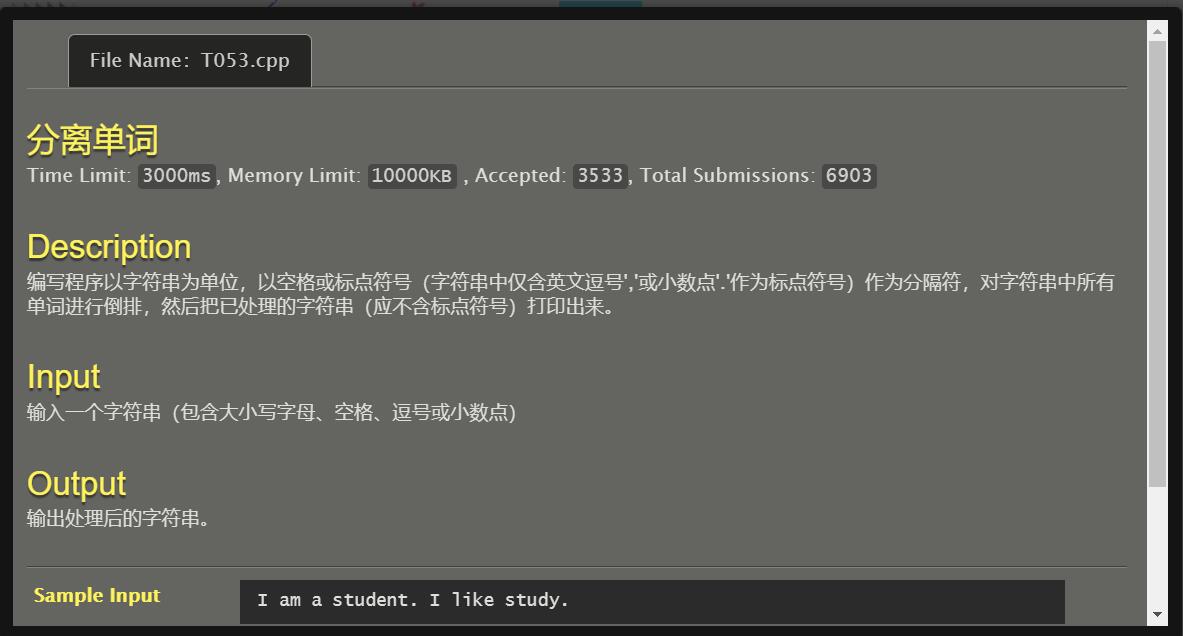
Left(s,n,dest);

Right(s,n,dest);

Mid(s,loc,n,dest);

return 0;

}



#include<stdio.h>

#include<string.h>

int letter\_flag(char c)

{

int flag=0;

if (((c>='a')&&(c<='z'))||((c>='A')&&(c<='Z'))) flag=1;

return(flag);

}

int main()

{

char s[10005];

int slen,i,j,k;

gets(s);

slen=strlen(s);

i=slen-1;

while (i>=0)

{

if (letter\_flag(s[i])==1)

{

j=i;

while (letter\_flag(s[i])==1) i--;

for (k=i+1;k<=j;k++) printf("%c",s[k]);

printf(" ");

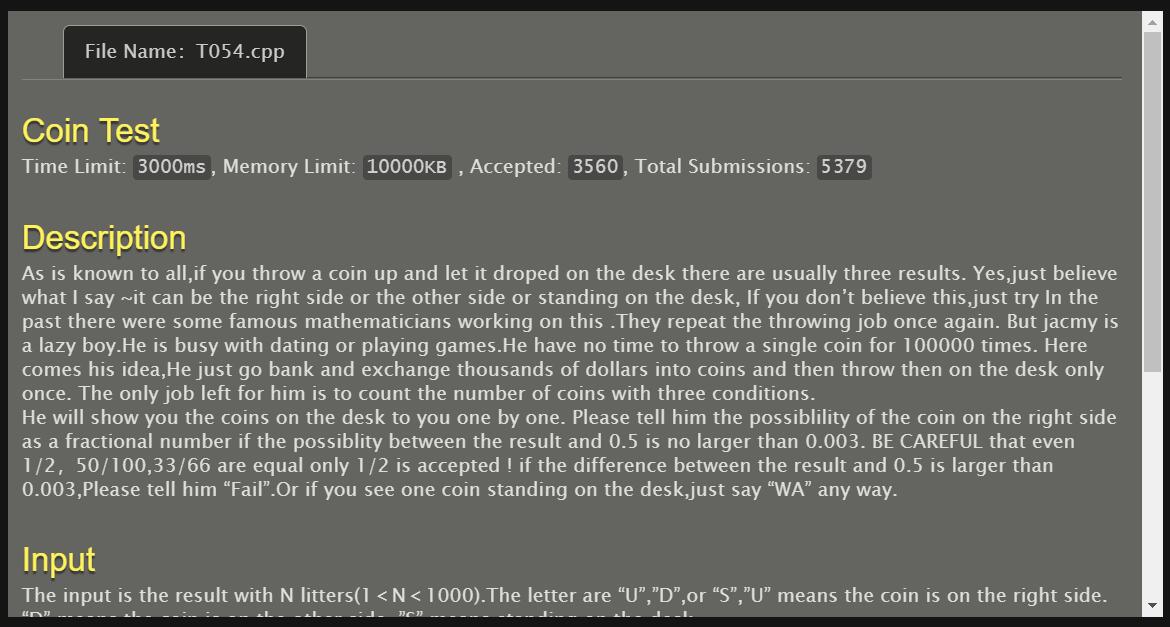
}

i--;

}

return 0;

}



#include<stdio.h>

#include<string.h>

#include<math.h>

int main()

{

char s[1005];

int len,i,cntu=0,cntd=0,cnts=0,sum=0;

double ans;

gets(s);

len=strlen(s);

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

{

if (s[i]=='U') cntu++;

else if (s[i]=='D') cntd++;

else cnts++;

}

if (cnts>=1) printf("WA\n");

else

{

sum=cntu+cntd;

ans=((1.0\*cntu)/(1.0\*sum));

if ((fabs(ans-0.5))<=0.003) printf("1/2\n");

else printf("Fail\n");

}

return 0;

}



#include<stdio.h>

#include<string.h>

void QuickSort1(char A[10005],int l,int r)

{

char temp;

int k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (A[++i]>A[k]));

while ((j>=l) && (A[--j]<A[k]));

if (i>=j) break;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

temp=A[k];

A[k]=A[j];

A[j]=temp;

QuickSort1(A,l,j-1);

QuickSort1(A,j+1,r);

}

}

void QuickSort2(char A[10005],int l,int r)

{

char temp;

int k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (A[++i]<A[k]));

while ((j>=l) && (A[--j]>A[k]));

if (i>=j) break;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

temp=A[k];

A[k]=A[j];

A[j]=temp;

QuickSort2(A,l,j-1);

QuickSort2(A,j+1,r);

}

}

int main()

{

char s[10005];

int len,mid,i;

gets(s);

len=strlen(s);

mid=(len/2);

if ((len%2)!=0)

{

QuickSort1(s,0,mid-1);

QuickSort2(s,mid+1,len-1);

for (i=mid+1;i<len;i++) printf("%c",s[i]);

printf("%c",s[mid]);

for (i=0;i<mid;i++) printf("%c",s[i]);

printf("\n");

}

else

{

QuickSort1(s,0,mid-1);

QuickSort2(s,mid,len-1);

for (i=mid;i<len;i++) printf("%c",s[i]);

for (i=0;i<mid;i++) printf("%c",s[i]);

printf("\n");

}

return 0;

}



#include<stdio.h>

#include<string.h>

void QuickSort(char A[10][10],int l,int r)

{

char temp[10];

int k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (strcmp(A[++i],A[k])<0));

while ((j>=l) && (strcmp(A[--j],A[k])>0));

if (i>=j) break;

strcpy(temp,A[i]);

strcpy(A[i],A[j]);

strcpy(A[j],temp);

}

strcpy(temp,A[k]);

strcpy(A[k],A[j]);

strcpy(A[j],temp);

QuickSort(A,l,j-1);

QuickSort(A,j+1,r);

}

}

int main()

{

char s[10][10];

int i;

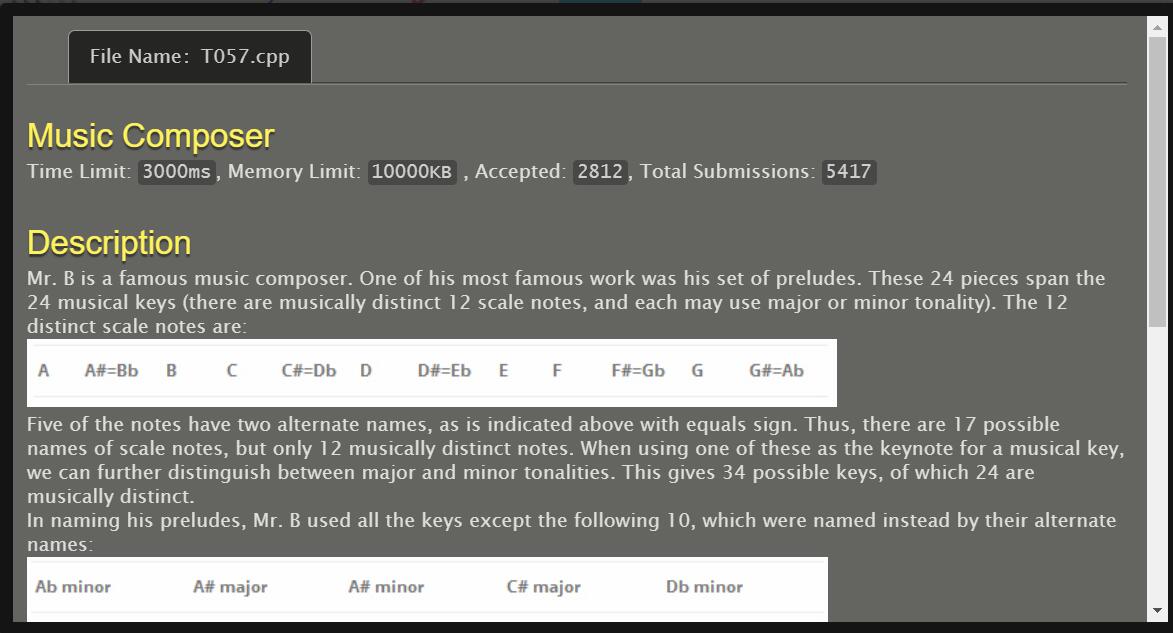
for (i=0;i<10;i++) scanf("%s",s[i]);

QuickSort(s,0,9);

for (i=0;i<10;i++) printf("%s ",s[i]);

return 0;

}



#include<stdio.h>

int main()

{

char s[20];

gets(s);

if ((s[0]=='A')&&(s[1]=='#')) {s[0]='B';s[1]='b';printf("%s\n",s);}

else if ((s[0]=='B')&&(s[1]=='b')) {s[0]='A';s[1]='#';printf("%s\n",s);}

else if ((s[0]=='C')&&(s[1]=='#')) {s[0]='D';s[1]='b';printf("%s\n",s);}

else if ((s[0]=='D')&&(s[1]=='b')) {s[0]='C';s[1]='#';printf("%s\n",s);}

else if ((s[0]=='D')&&(s[1]=='#')) {s[0]='E';s[1]='b';printf("%s\n",s);}

else if ((s[0]=='E')&&(s[1]=='b')) {s[0]='D';s[1]='#';printf("%s\n",s);}

else if ((s[0]=='F')&&(s[1]=='#')) {s[0]='G';s[1]='b';printf("%s\n",s);}

else if ((s[0]=='G')&&(s[1]=='b')) {s[0]='F';s[1]='#';printf("%s\n",s);}

else if ((s[0]=='G')&&(s[1]=='#')) {s[0]='A';s[1]='b';printf("%s\n",s);}

else if ((s[0]=='A')&&(s[1]=='b')) {s[0]='G';s[1]='#';printf("%s\n",s);}

else printf("UNIQUE\n");

return 0;

}



#include<stdio.h>

#include<string.h>

int main()

{

char s[105];

int i,m,len;

gets(s);

scanf("%d",&m);

len=strlen(s);

for (i=m;i<len;i++) printf("%c",s[i]);

printf("\n");

return 0;

}



#include<stdio.h>

#include<string.h>

int main()

{

char s1[300],s2[100],s3[100],c;

int i,LA=0,La=0,num=0,space=0,others=0;

gets(s1);

gets(s2);

gets(s3);

strcat(s1,s2);

strcat(s1,s3);

for (i=0;s1[i]!='\0';i++)

{

c=s1[i];

if ((c>='A')&&(c<='Z')) LA++;

else if ((c>='a')&&(c<='z')) La++;

else if ((c>='0')&&(c<='9')) num++;

else if (c==' ') space++;

else others++;

}

printf("%d %d %d %d %d\n",LA,La,num,space,others);

return 0;

}



#include<stdio.h>

#include<string.h>

void stringmerge(char s1[],const char s2[])

{

int i,lensum,len1,len2;

len1=strlen(s1);

len2=strlen(s2);

lensum=len1+len2;

for (i=len1;i<lensum;i++) s1[i]=s2[i-len1];

s1[i]='\0';

puts(s1);

}

int main()

{

char s1[10005],s2[10005];

gets(s1);

gets(s2);

stringmerge(s1,s2);

return 0;

}



#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int min(int x,int y)

{

if (x>y) return(y);

else return(x);

}

int main()

{

char s[10];

char \*p=s,\*q=s,\*m=s,\*n=s;

int i,a,b,c,d,sumad,sumbc,sumbd,sum1,sum2,mincnt,flag=0;

gets(s);

p=&s[0];

q=&s[2];

m=&s[4];

n=&s[6];

a=atoi(p);

b=atoi(q);

c=atoi(m);

d=atoi(n);

sumad=(a\*d);

sumbc=(b\*c);

sumbd=(b\*d);

if (s[3]=='+')

{

sum1=sumad+sumbc;

mincnt=min(sum1,sumbd);

for (i=mincnt;i>=2;i--) if ((sum1%i==0)&&(sumbd%i==0))

{

sum1=(sum1/i);

sumbd=(sumbd/i);

break;

}

if ((i==mincnt)&&(sum1>=sumbd)) printf("%d\n",(sum1/sumbd));

else printf("%d/%d\n",sum1,sumbd);

}

else

{

sum2=sumad-sumbc;

if (sum2==0) printf("0\n");

else

{

if (sum2<0)

{

flag=1;

sum2=-sum2;

}

mincnt=min(sum2,sumbd);

for (i=mincnt;i>=2;i--) if ((sum2%i==0)&&(sumbd%i==0))

{

sum2=(sum2/i);

sumbd=(sumbd/i);

break;

}

if (flag==1) printf("-");

if ((i==mincnt)&&(sum2>=sumbd)) printf("%d\n",(sum2/sumbd));

else printf("%d/%d\n",sum2,sumbd);

}

}

return 0;

}



#include<stdio.h>

int main()

{

char s1[1005],s2[1005];

int cnt=0;

FILE \*in,\*out;

in=fopen("DATA5612.CPP","r");

if (in!=NULL)

{

out=fopen("DATA5612.TXT","w");

while (!feof(in))

{

if (fgets(s1,sizeof(s1)-1,in)==NULL) continue;

sprintf(s2,"%04d %s",++cnt,s1);

fputs(s2,out);

}

fclose(out);

fclose(in);

}

return 0;

}



#include<stdio.h>

#include<stdlib.h>

#include<string.h>

void fun(const char \*s,char \*t)

{

int i,j=0;

for (i=0;s[i]!='\0';i++)

{

if (((i%2)!=0)&&((s[i]%2)!=0)) t[j++]=s[i];

}

t[j]='\0';

puts(t);

}

int main()

{

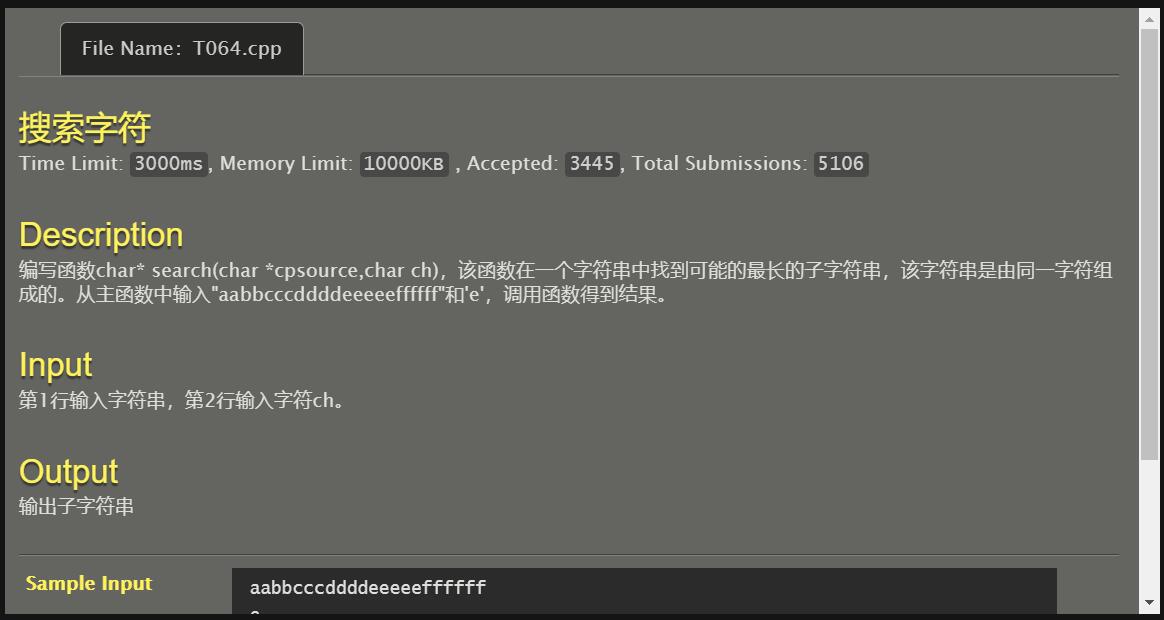
char s1[1005],s2[1005],\*s=s1,\*t=s2;

gets(s1);

fun(s,t);

return 0;

}



#include<stdio.h>

#include<stdlib.h>

#include<string.h>

char s1[10005],s2[10005],\*p=s2;

int maxstr=0;

char \*search(char \*cpsource,char ch)

{

int i=0,j,slen,cnt;

slen=strlen(cpsource);

while (i<slen)

{

if ((cpsource[i]-ch)==0)

{

j=0;

cnt=0;

while ((cpsource[i]-ch)==0)

{

p[j++]=cpsource[i];

cnt++;

i++;

}

if (cnt>maxstr) maxstr=cnt;

}

i++;

}

return p;

}

int main()

{

char ch;

int k;

gets(s1);

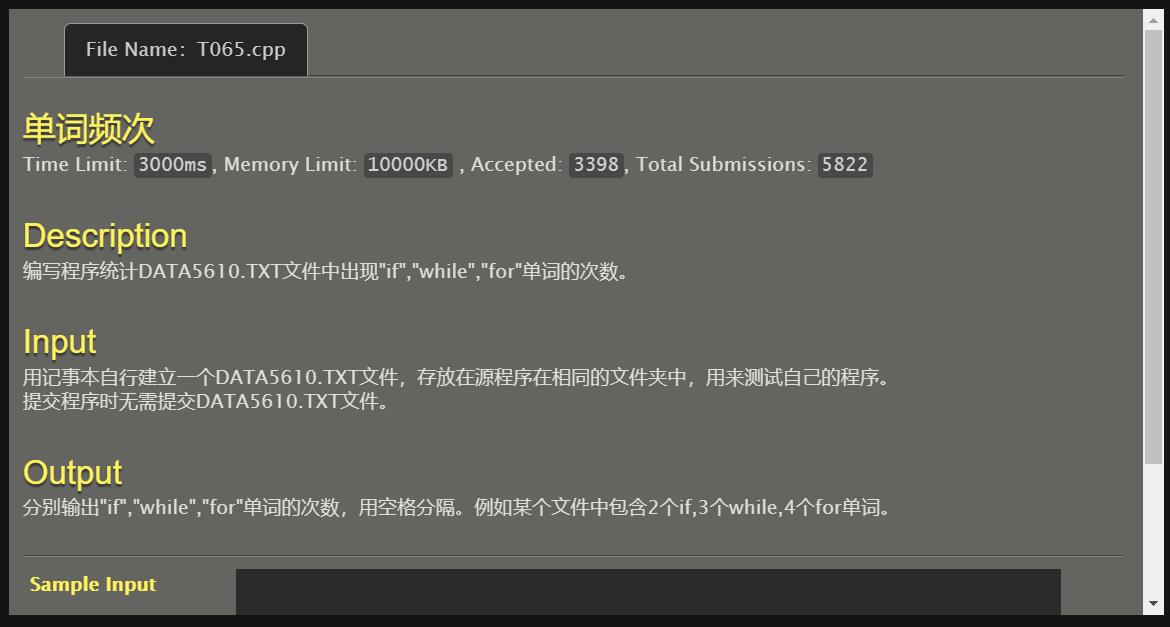
scanf("%c",&ch);

search(s1,ch);

for (k=0;k<maxstr;k++) printf("%c",s2[k]);

return 0;

}



#include<stdio.h>

#include<string.h>

int main()

{

char s[10005];

int i,cnt\_if=0,cnt\_while=0,cnt\_for=0;

FILE \*in;

in=fopen("DATA5610.TXT","r");

if (in!=NULL)

{

while (!feof(in))

{

if (fgets(s,sizeof(s)-1,in)==NULL) continue;

i=0;

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

{

if ((s[i]=='i')&&(s[i+1]=='f'))

{

cnt\_if++;

i=i+2;

}

else if ((s[i]=='w')&&(s[i+1]=='h')&&(s[i+2]=='i')&&(s[i+3]=='l')&&(s[i+4]=='e'))

{

cnt\_while++;

i=i+5;

}

else if ((s[i]=='f')&&(s[i+1]=='o')&&(s[i+2]=='r'))

{

cnt\_for++;

i=i+3;

}

else i++;

}

}

fclose(in);

}

printf("%d %d %d\n",cnt\_if,cnt\_while,cnt\_for);

return 0;

}



#include<stdio.h>

#include<stdlib.h>

int main()

{

unsigned int t;

int n,flag=0;

scanf("%d",&n);

if (n==0) printf("0");

else

{

for (t=1<<31;t>0;t=t>>1)

{

if (((t&n)==0) && (flag==0)) continue;

if (((t&n)==0) && (flag==1)) printf("0");

if (t&n)

{

printf("1");

flag=1;

}

}

}

printf("\n");

return 0;

}



#include<stdio.h>

#include<string.h>

typedef struct Student{

int number;

char name[18];

int score;

}Student;

void QuickSort(Student A[],int l,int r)

{

Student temp;

int k,i,j;

if (l<r)

{

i=l;j=r+1;

k=l;

while (1)

{

while ((i<r) && (A[++i].score>A[k].score));

while ((j>=l)&& (A[--j].score<A[k].score));

if (i>=j) break;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

temp=A[k];

A[k]=A[j];

A[j]=temp;

QuickSort(A,l,j-1);

QuickSort(A,j+1,r);

}

}

int main()

{

Student A[1005],B[1005];

int n,i,cnt=0;

scanf("%d",&n);

for (i=0;i<n;i++) scanf("%d%s%d",&A[i].number,&A[i].name,&A[i].score);

FILE \*out,\*in;

out=fopen("DATA5614.DB","w");

for (i=0;i<n;i++) fprintf(out,"%d %s %d\n",A[i].number,A[i].name,A[i].score);

fclose(out);

in=fopen("DATA5614.DB","r");

if (in!=NULL)

{

while (!feof(in))

{

fscanf(in,"%d%s%d",&B[cnt].number,&B[cnt].name,&B[cnt].score);

cnt++;

}

fclose(in);

QuickSort(B,0,n-1);

}

printf("%d %s %d\n",B[0].number,B[0].name,B[0].score);

return 0;

}



#include<stdio.h>

#include<string.h>

void replace(char \*str,char \*newstr,const char \*fstr,const char \*rstr)

{

int i,j,lenstr,lenfstr,lenrstr;

lenstr=strlen(str);

lenfstr=strlen(fstr);

lenrstr=strlen(rstr);

i=0;

j=0;

while (i<lenstr)

{

if (strncmp(str+i,fstr,lenfstr)==0)

{

strcat(newstr+j,rstr);

j+=lenrstr;

i+=lenfstr;

}

else

{

newstr[j++]=str[i++];

}

}

//newstr[j]='\0';

}

int main()

{

char str[1005],fstr[1005],rstr[1005],newstr[1005];

gets(str);

gets(fstr);

gets(rstr);

replace(str,newstr,fstr,rstr);

puts(newstr);

return 0;

}



#include<stdio.h>

#include<string.h>

void strencode(char \*s)

{

while (\*s!='\0')

{

if ((\*s>='A') && (\*s<='Z')) \*s+=3;

if ((\*s>='a') && (\*s<='z')) \*s-=3;

s++;

}

}

int main()

{

char s[1005];

gets(s);

strencode(s);

puts(s);

return 0;

}



#include<stdio.h>

#include<string.h>

void oneinsert(char \*a,char \*anew,char x,int i)

{

strncpy(anew,a,i);

anew[i]=x;

anew[i+1]='\0';

strcat(anew,a+i);

}

int main()

{

char a[1005],anew[1005],x;

int i;

gets(a);

scanf("%c%d",&x,&i);

oneinsert(a,anew,x,i);

puts(anew);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

typedef struct tagLNode{

int data;

struct tagLNode \*next;

}LNode,\*LinkList;

void create(LinkList \*l,int n)

{

LinkList p,s;

p=\*l=(LinkList)malloc(sizeof(LNode));

for(;n>0;n--){

s=(LinkList)malloc(sizeof(LNode));

scanf("%d",&s->data);

p->next=s,p=s;

}

p->next=NULL;

}

void list(LinkList l)

{

LinkList p=l->next;

while(p!=NULL){

printf("%d ",p->data);

p=p->next;

}

}

void chazhao(LinkList l,int n)

{

LinkList p=l->next,s,q;

int i,j,k;

for(i=0;i<100;i++){

j=p->data;

q=p->next;

k=q->data;

if(j<n&&k>n) break;

p=p->next;}

s=(LinkList)malloc(sizeof(LNode));

s->data=n;

s->next=p->next,p->next=s;

}

int main()

{

LinkList q;

int n,m;

q=(LinkList)malloc(sizeof(LNode));

scanf("%d",&n);

create(&q,n);

scanf("%d",&m);

chazhao(q,m);

list(q);

return 0;

}



#include<stdio.h>

#include<stdlib.h>

typedef struct tagNode{

int data;

struct tagNode \*prev,\*next;

}Node,\*LinkList;

void CreateLink(LinkList \*L,int n)

{

LinkList p,q,s;

int u;

p=\*L=(LinkList)malloc(sizeof(Node));

p->prev=NULL;

s=(LinkList)malloc(sizeof(Node));

p->next=s,s->prev=p,s->next=NULL;

scanf("%d",&s->data);

for(;n>1;n--){

u=0;

p=(\*L)->next;

s=(LinkList)malloc(sizeof(Node));

scanf("%d",&s->data);

while(p!=NULL){

if(p->data==s->data){

if(p->next==NULL){

p->prev->next=NULL;

free(p),free(s),p=NULL;

}

else{

q=p->prev->next=p->next;

p->next->prev=p->prev;

free(p),p=q,u++;

}

}

else{

if(p->next!=NULL) p=p->next;

else{

if(u!=0) p=p->next,free(s);

else{

p->next=s,s->prev=p,p=s->next=NULL;

}

}

}

}

}

}

int main()

{

LinkList L;

int n;

scanf("%d",&n);

CreateLink(&L,n);

while(L->next!=NULL){

L=L->next;

printf("%d ",L->data);

}

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <assert.h>

void add(const char \*num1, const char \*num2, char \*result)

{

if(!result) return;

if(!num1) { strcpy(result, num2); return; }

if(!num2) { strcpy(result, num1); return; }

char mark;

int bits1, bits2;

char \*temp\_str = NULL;

char \*temp\_str2 = NULL;

int temp, max, min, a;

if(num1[0] == '-' && num2[0] != '-')

{

bits1 = strlen(num1) - 1;

bits2 = strlen(num2);

max = bits1 > bits2 ? bits1 : bits2;

min = bits1 < bits2 ? bits1 : bits2;

if(bits1 > bits2) mark = 1;

else if(bits1 < bits2) mark = 0;

else

{

temp = strcmp(num1+1, num2);

if(temp > 0) mark = 1;

else if(temp < 0) mark = 0;

else { result = "0"; return; }

}

if(mark == 1)

{

max++;

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

strcpy(temp\_str, num1);

while(min-- > 0)

{

a = temp\_str[--max] - num2[--bits2];

if(a < 0)

{

temp\_str[max - 1]--;

a += 10;

}

temp\_str[max] = a + '0';

}

temp\_str2 = temp\_str;

while(\*(++temp\_str) == '0') ;

if(\*temp\_str == '\0') result = "0";

else

{

strcpy(result+1, temp\_str);

result[0] = '-';

}

temp\_str = temp\_str2;

}

else

{

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

strcpy(temp\_str, num2);

while(min-- > 0)

{

a = temp\_str[--max] - num1[bits1--];

if(a < 0)

{

temp\_str[max - 1]--;

a += 10;

}

temp\_str[max] = a + '0';

}

temp\_str2 = temp\_str;

while(\*temp\_str == '0') temp\_str++;

if(\*temp\_str == '\0') result = "0";

else strcpy(result, temp\_str);

temp\_str = temp\_str2;

}

if(temp\_str) free(temp\_str);

return;

}

else if(num1[0] != '-' && num2[0] == '-')

{

bits1 = strlen(num1);

bits2 = strlen(num2) - 1;

max = bits1 > bits2 ? bits1 : bits2;

min = bits1 < bits2 ? bits1 : bits2;

if(bits1 > bits2) mark = 0;

else if(bits1 < bits2) mark = 1;

else

{

temp = strcmp(num1, num2+1);

if(temp > 0) mark = 0;

else if(temp < 0) mark = 1;

else { result = "0"; return; }

}

if(mark == 1)

{

max++;

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

strcpy(temp\_str, num2);

while(min-- > 0)

{

a = temp\_str[--max] - num1[--bits1];

if(a < 0)

{

temp\_str[max - 1]--;

a += 10;

}

temp\_str[max] = a + '0';

}

temp\_str2 = temp\_str;

while(\*(++temp\_str) == '0') ;

if(\*temp\_str == '\0') result = "0";

else

{

strcpy(result+1, temp\_str);

result[0] = '-';

}

temp\_str = temp\_str2; }

else

{

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

strcpy(temp\_str, num1);

while(min-- > 0)

{

a = temp\_str[--max] - num2[bits2--];

if(a < 0)

{

temp\_str[max - 1]--;

a += 10;

}

temp\_str[max] = a + '0';

}

temp\_str2 = temp\_str;

while(\*temp\_str == '0') temp\_str++;

if(\*temp\_str == '\0') result = "0";

else strcpy(result, temp\_str);

temp\_str = temp\_str2;

}

if(temp\_str) free(temp\_str);

return ;

}

else if(num1[0] == '-' && num2[0] == '-')

{

bits1 = strlen(num1) - 1;

bits2 = strlen(num2) - 1;

max = bits1 > bits2 ? bits1 : bits2;

min = bits1 < bits2 ? bits1 : bits2;

max += 2;

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

if(bits1 > bits2)

{

strcpy(temp\_str+2, num1+1);

while(min-- > 0)

{

a = (temp\_str[--max] - '0') + (num2[bits2--] - '0');

if(a >= 10)

{

temp\_str[max - 1]++;

a -= 10;

}

temp\_str[max] = a + '0';

}

}

else

{

strcpy(temp\_str+2, num2+1);

while(min-- > 0)

{

a = (temp\_str[--max] - '0') + (num1[bits1--] - '0');

if(a >= 10)

{

temp\_str[max - 1]++;

a -= 10;

}

temp\_str[max] = a + '0';

}

}

temp\_str2 = temp\_str;

while(\*(++temp\_str) == '0') ;

if(\*temp\_str == '\0') result = "0";

else

{

strcpy(result+1, temp\_str);

result[0] = '-';

}

if(temp\_str2) free(temp\_str2);

return;

}

else

{

bits1 = strlen(num1);

bits2 = strlen(num2);

max = bits1 > bits2 ? bits1 : bits2;

min = bits1 < bits2 ? bits1 : bits2;

max++;

temp\_str = (char \*) malloc(max+1);

assert(temp\_str != NULL);

memset(temp\_str, '0', max+1);

if(bits1 > bits2)

{

strcpy(temp\_str+1, num1);

while(min-- > 0)

{

a = (temp\_str[--max] - '0') + (num2[--bits2] - '0');

if(a >= 10)

{

temp\_str[max - 1]++;

a -= 10;

}

temp\_str[max] = a + '0';

}

}

else

{

strcpy(temp\_str+1, num2);

while(min-- > 0)

{

a = (temp\_str[--max] - '0') + (num1[--bits1] - '0');

if(a >= 10)

{

temp\_str[max - 1]++;

a -= 10;

}

temp\_str[max] = a + '0';

}

}

temp\_str2 = temp\_str;

while(\*temp\_str == '0') temp\_str++;

if(\*temp\_str == '\0') result = "0";

else strcpy(result, temp\_str);

if(temp\_str2) free(temp\_str2);

return ;

}

}

int main()

{

char a[100],b[100],c[100];

gets(b);

gets(c);

add(b,c,a);

printf("%s",a);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

char a[100];

int i,h,j;

gets(a);

for(i=0;i<strlen(a);i++)

{

h=0;

for(j=i;j<strlen(a)&&h<9;j++)

if(a[j]==a[i]) h++;

else break;

printf("%d%c",h,a[i]);

i=i+h-1;

}

return 0;

}



#include<stdio.h>

#include<string.h>

int main()

{

char a[102],b[102],\*p=a,\*q=b,\*r;

int i,j,m,n,u=0,v=0,f[200];

gets(a);

gets(b);

m=strlen(a),n=strlen(b);

if(a[0]=='-'){

u++;

for(i=0;i<m;i++)a[i]=a[i+1];

m-=1;}

if(b[0]=='-'){

v++;

for(i=0;i<n;i++)b[i]=b[i+1];

n-=1;}

if((u==0&&v==0)||(u==1&&v==1)){

if(u==0&&v==0){

if(m<n) r=p,p=q,q=r,u=m,m=n,n=u,printf("-");

if(m==n){for(i=0;i<m;i++)if(a[i]<b[i]){printf("-");

r=p,p=q,q=r;

break;}}}

else {

if(m<n) r=p,p=q,q=r,u=m,m=n,n=u;

else if(m==n){for(i=0;i<m;i++)if(a[i]<b[i]){

r=p,p=q,q=r;

break;}

if(i==m) printf("-");}

else printf("-");}

for(i=m-1,u=n-1;i>=m-n;i--,u--)q[i]=q[u];

for(i=0;i<m-n;i++)q[i]=48;

for(i=m-1;i>=0;i--){

if(p[i]>=q[i]) f[i]=p[i]-q[i];

else{

u=i;

if(p[u-1]==48){

f[i]=p[i]+10-q[i];

while(p[u-1]==48){

p[u-1]='9';

if(p[u-2]!=48)p[u-2]-=1;

u--;}}

else f[i]=p[i]+10-q[i],p[i-1]-=1;}}

for(i=0;;i++)if(f[i]!=0)break;

for(;i<m;i++)printf("%d",f[i]);}

else{

if(u==1)printf("-");

if(m<n) r=p,p=q,q=r,u=m,m=n,n=u;

for(i=m-1,j=n-1;i>=m-n;i--,j--)q[i]=q[j];

for(i=0;i<m-n;i++)q[i]=48;

for(i=m-1,j=0;i>=0;i--){

f[i]=(p[i]-48+q[i]-48+j)%10;

j=(p[i]-48+q[i]-48+j)/10;}

if(j!=0){ printf("%d",j);

for(i=0;i<m;i++)printf("%d",f[i]);}

else for(i=0;i<m;i++)printf("%d",f[i]);}

return 0;

}



#include<stdio.h>

#include<stdlib.h>

typedef struct tagNode{

int data;

struct tagNode \*next;

}Node,\*LinkList;

void CreateLink(LinkList \*L,int n)

{

LinkList p,s;

int i;

p=(\*L)=(LinkList)malloc(sizeof(Node));

p->data=n;

for(i=1;n>0;n--,i++){

s=(LinkList)malloc(sizeof(Node));

s->data=i;

p->next=s,p=s;

}

p->next=NULL;

}

int main()

{

int i,j,m,n,x,a[20];

LinkList L,p,s,t;

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

CreateLink(&L,n);

for(i=0;i<20;i++) scanf("%d",&a[i]);

for(i=0;L->data!=x;i++){

m=L->data;

p=L,t=s=L->next;

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

if(j%a[i]==0){

t=p->next=s->next;

free(s),s=t;

(L->data)--;

if((L->data)==x)break;

}

else p=s,s=s->next;

}

}

p=L->next;

while(p!=NULL){

printf("%d ",p->data);

p=p->next;

}

}



#include<stdio.h>

int main()

{

int i,j=0,k,t,n,a[1000];

scanf("%d",&n);

for(i=0;i<n;i++)scanf("%d",&a[i]);

for(;;)

{

for(i=0;i<n-1;i++)

{

if(a[i]>a[i+1])t=a[i],a[i]=a[i+1],a[i+1]=t,j++;

}

for(k=0;k<n-1;k++)

{

if(a[k]>a[k+1])break;

}

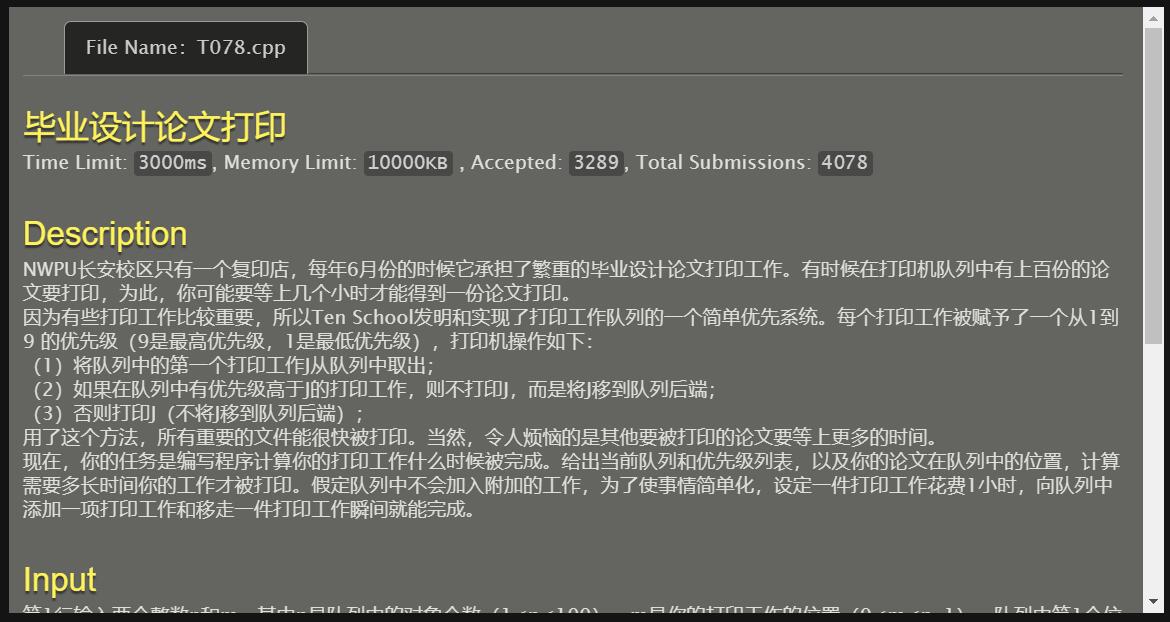
if(k==n-1)break;

}

printf("%d",j);

return 0;

}



#include<stdio.h>

int main()

{

int n,m,i,cnt=1,a[100];

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

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

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

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

if(a[i]>a[m]) cnt++;

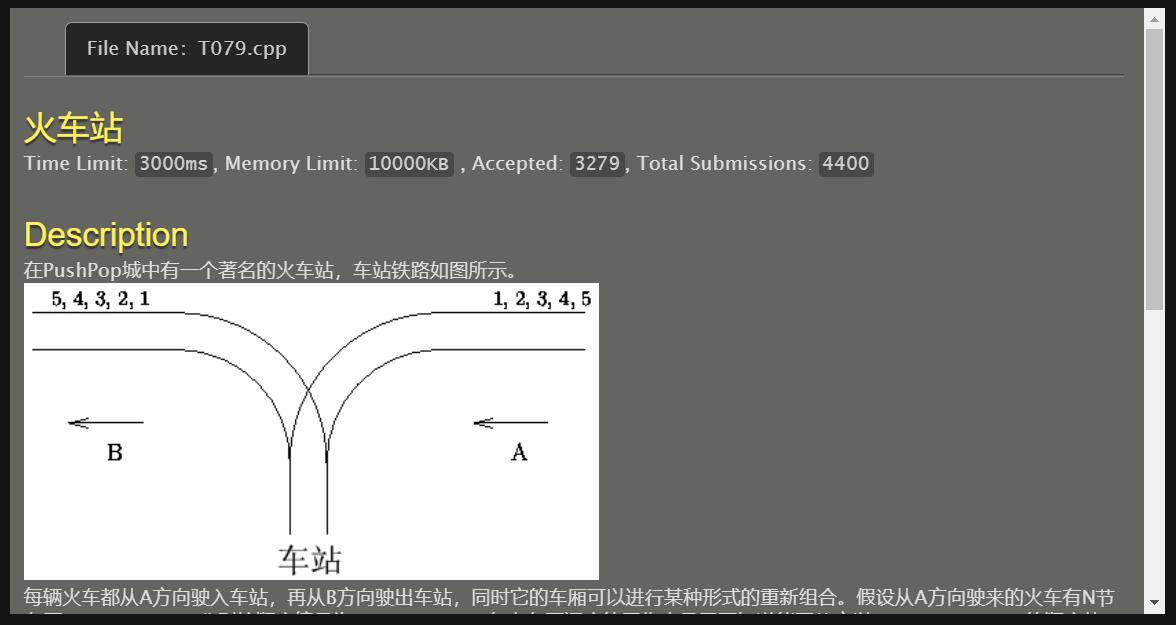
for(i=0;i<m;i++)

if(a[i]==a[m]) cnt++;

printf("%d\n",cnt);

return 0;

}



#include<stdio.h>

#include<stdlib.h>

typedef struct tagNode{

int data;

struct tagNode \*prev,\*next;

}Node,\*LinkList;

int Judge(int a[]){

int i=0,j=1;

LinkList L,p=L=(LinkList)malloc(sizeof(Node)),s;

while(a[i]!=0){if(j==a[i])j++,i++;

if(a[i]==0)break;

if(j<a[i]){for(;j!=a[i];j++){

s=(LinkList)malloc(sizeof(Node));

s->data=j,s->prev=p,s->next=NULL,p->next=s,p=s;}

i++,j++;}

if(a[i]==0)break;

if(j>a[i]){if(p->data!=a[i])return 0;

else {p->prev->next=NULL,s=p->prev;

free(p),p=s;}

i++;}

if(a[i]==0)break;}

if(a[i]==0)return 1;

}

int main()

{

int i,n,a[1000];

scanf("%d",&n);

for(i=0;i<=n;i++)scanf("%d",&a[i]);

if(Judge(a)==1)printf("Yes\n");

else printf("No\n");

return 0;

}



//需修改！

#include<stdio.h>

int main()

{

int i,n,a[10000];

scanf("%d",&n);

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

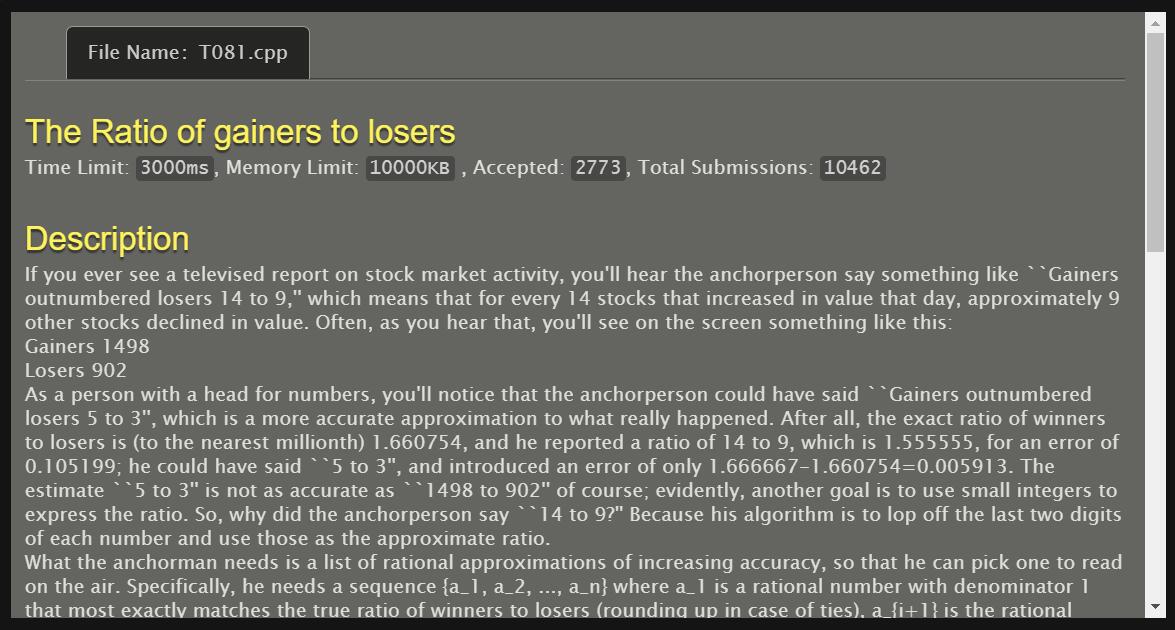
scanf("%d",&a[i]);

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

printf("%d ",a[i]);

return 0;

}



#include <stdio.h>

#include <math.h>

inline int rounding(double a)

{

int b;

b=(int)a;

if(a-b<0.5) return b;

else return ++b;

}

int main()

{

int a,b,x,xx=9999,ii=1;

double t;

scanf("%d %d",&a,&b);

while(1)

{

t=a\*1.0/b;

int i=0;

double e=9999,f,g=1;

do

{

i++;

x=rounding(t\*i);

if((f=fabs(x\*1.0/i-t))<e)

{

xx=x;ii=i;e=f;g=xx\*b-ii\*a;

printf("%d/%d",xx,ii);

printf("\n");

}

}

while(g);

xx=9999;ii=1;

break;

}

return 0;

}



#include<stdio.h>

int main()

{

long long int a,b;

scanf("%lld%lld",&a,&b);

if(a==0||b==0)

printf("0\n");

else

{

long long int n=((a%9)\*(b%9))%9;

if(n==0) n=9;

printf("%lld\n",n);

}

}



#include <stdio.h>

int fun(int n)

{

int i,m=1,sum=0;

for(i=1;i<12;i++)

{

m\*=i;

sum+=m;

if(n==m) return 1;

if(m>n) break;

}

if(sum-m<n) return 0;

else return fun(n-m/i);

}

int main()

{

int n;

scanf("%d",&n);

if(fun(n)) printf("YES\n");

else printf("NO\n");

return 0;

}



#include <stdio.h>

#include <stdlib.h>

int main()

{

float c,n=0;

double i=1;

scanf("%f",&c);

while(n<c){

n=n+1/(i+1);

i++;

}

printf("%.0lf",i-1);

return 0;

}



#include<stdio.h>

int main()

{

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

scanf("%d%d%d%d%d%d",&x1,&y1,&x2,&y2,&x3,&y3);

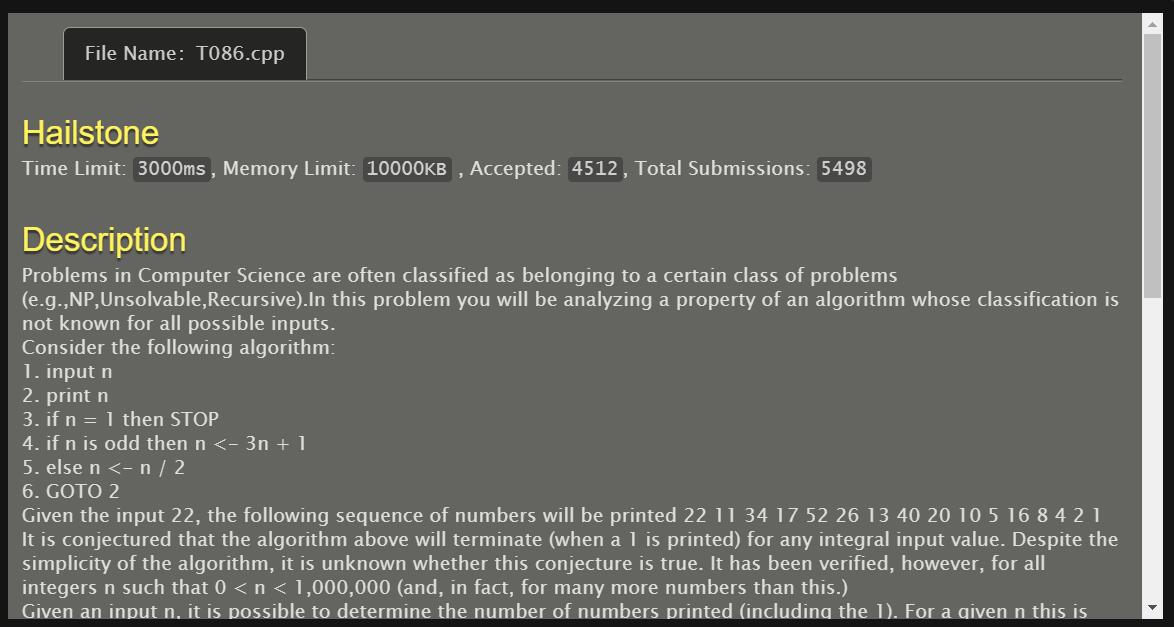
if((x2-x1)\*(y3-y1)-(x3-x1)\*(y2-y1)>0)

printf("0\n");

else printf("1\n");

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int f(int n)

{

int s=1;

while(1){

if(n==1) return s;

else if(n%2==0) n=n/2,s++;

else n=3\*n+1,s++;

}

}

int main()

{

int n,m,i,j=0,t;

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

printf("%d %d",m,n);

if(m>n) t=m,m=n,n=t;

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

if(f(i)>j) j=f(i);

printf(" %d",j);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

int main()

{

int n,i,s=0;

scanf("%d",&n);

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

s=s+10;

if(s>=n) break;

s=s-5;

}

printf("%d",i+1);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

int a[100],i,n,left=0,right=0;

char c[5];

scanf("%s%d",c,&n);

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

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

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

if(a[i]==0) left++;

right=n-left;

n=right-left;

if(n>=0) n=n%4;

else n=4-(-n)%4;

if(c[0]=='E'){

if(n==0) printf("East");

if(n==1) printf("South");

if(n==2) printf("West");

if(n==3) printf("North");

}

if(c[0]=='S'){

if(n==3) printf("East");

if(n==0) printf("South");

if(n==1) printf("West");

if(n==2) printf("North");

}

if(c[0]=='W'){

if(n==2) printf("East");

if(n==3) printf("South");

if(n==0) printf("West");

if(n==1) printf("North");

}

if(c[0]=='N'){

if(n==1) printf("East");

if(n==2) printf("South");

if(n==3) printf("West");

if(n==0) printf("North");

}

return 0;

}



#include<stdio.h>

int main()

{

int i,j,k=0,h,n,m,a[100],t,o=0,q;

scanf("%d",&m);

scanf("%d",&n);

for(i=0;i<n;i++) scanf("%d",&a[i]);

for(i=0;i<n-1;i++)

for(j=0;j<n-1-i;j++)

if(a[j]<a[j+1]) t=a[j],a[j]=a[j+1],a[j+1]=t;

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

{

q=0;

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

if(a[i]+a[j]<=m)

{

k++;o++;q++;

for(h=j;h<n-o-1;h++) a[h]=a[h+1];

break;

};

if(q==0) k++;

}

printf("%d",k);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void BubbleSort(double A[],int s,int m)

{

int i,j;

double k;

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

for(i=s;i<m+s-1;i++)

if(A[i]>A[i+1])

{

k=A[i];

A[i]=A[i+1];

A[i+1]=k;

}

}

int main()

{

int n,i;

double X[10000],Y[10000],s1=0,s2=0,s=0;

scanf("%d",&n);

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

{

scanf("%lf%lf",&X[i],&Y[i]);

s2+=X[i];

}

BubbleSort(X,0,n);

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

s1+=X[i]-i\*1.0;

s1=(int)((s1/n) + 0.5);

s2=(int)((s2/n) + 0.5);

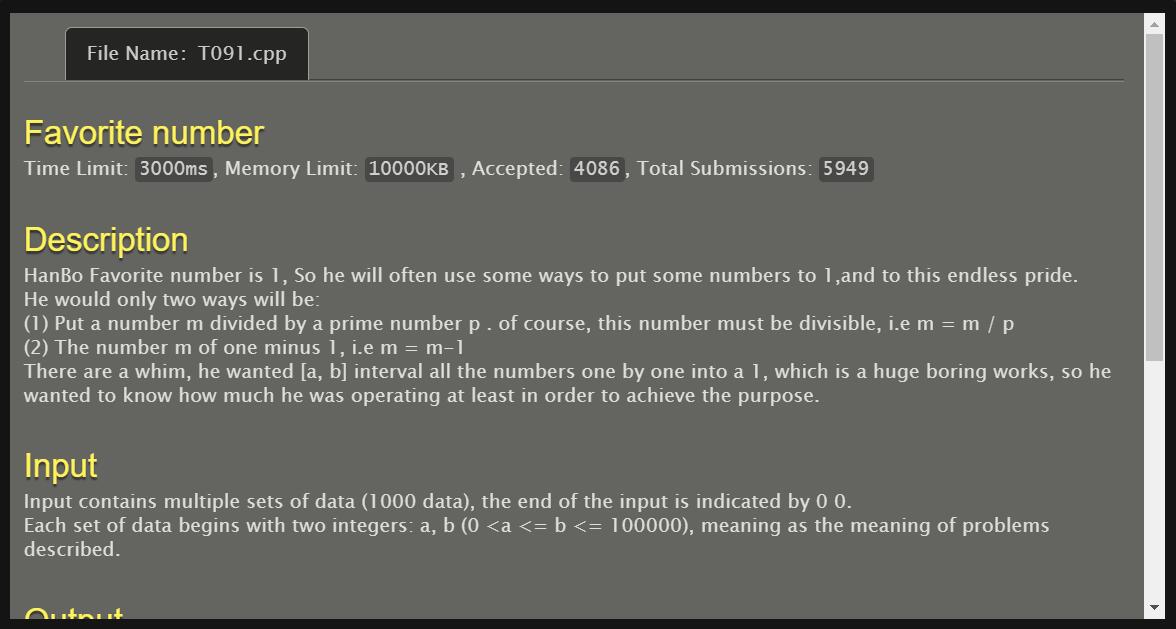
for(i=0;i<n;i++)

s=s+abs(X[i]-i-s1)+abs(Y[i]-s2);

printf("%.0lf",s);

return 0;

}



#include<stdio.h>

#include<math.h>

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

int prime[10000];

int cnt;

int isprime[100010]={0};

int dp[100010]={0},sum[100010]={0};

int countmin(int x)

{

int i,lim=sqrt(x)+1;

if(dp[x]!=0)return dp[x];

dp[x]=dp[x-1]+1;

int xx=x;

for(i=0;i<cnt&&prime[i]<=lim&&xx!=1;i++)

{

if(x%prime[i]==0)

{

while(xx%prime[i]==0)xx/=prime[i];

dp[x]=min(dp[x],dp[x/prime[i]]+1);

}

}

if(xx!=1)dp[x]=min(dp[x],dp[x/xx]+1);

return dp[x];

}//得出m->1的最少操作次数

void calprime()

{

int i=2,j;

for(i=2;i<=100000;i++)

isprime[i]=1;

cnt=0;

for(i=2;i<=100000;i++)

{

if(isprime[i])

{

prime[cnt]=i;

cnt++;

for(j=i\*i;i<400&&j<=100000;j+=i)

isprime[j]=0;

}

}//筛法求出100000以内的所有素数；

for(i=2;i<=100000;i++)countmin(i);

}

int main()

{

int a,b,i;

calprime();

for(i=2;i<=100000;i++)sum[i]=sum[i-1]+dp[i];

while(scanf("%d%d",&a,&b)!=EOF&&a!=0&&b!=0)

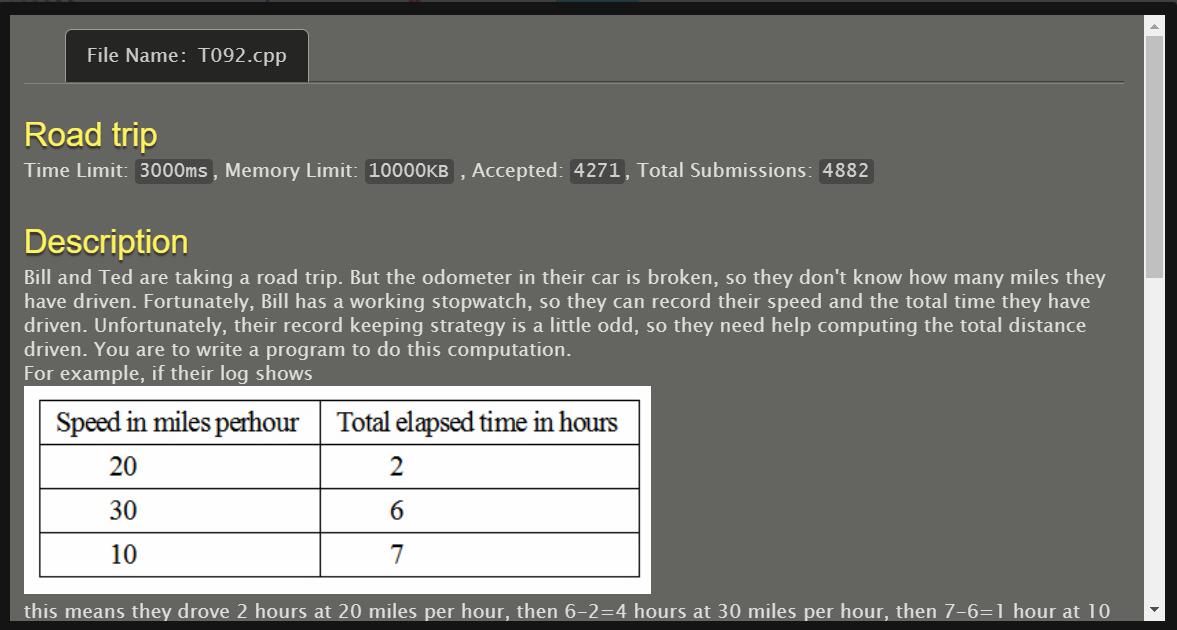
{

printf("%d\n",sum[b]-sum[a-1]);

}

return 0;

}



#include<stdio.h>

#include<stdlib.h>

int main()

{

int i,j,n,\*p,b[1000],cnt=0;

scanf("%d",&n);

while(n!=-1){b[cnt]=0,p=(int\*)malloc(n\*2\*sizeof(int));

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

for(j=0;j<2;j++)scanf("%d",p+i\*2+j);

b[cnt]=p[0]\*p[1];

for(i=1;i<n;i++) b[cnt]=b[cnt]+(p[i+i+1]-p[i+i-1])\*p[i+i];

cnt++,free(p),scanf("%d",&n);}

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

}



#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int len[64],n,minlen,get;

int used[64];

void quickSort(int a[],int s,int m)

{

int i,j,t;

if(s<m){

i=s,j=m+1;

while(1)

{

while(i+1<64&&a[++i]>a[s]);

while(j-1>-1&&a[--j]<a[s]);

if(i>=j)break;

t=a[i],a[i]=a[j],a[j]=t;

}

t=a[s],a[s]=a[j],a[j]=t;

quickSort(a,s,j-1);

quickSort(a,j+1,m);}}

//现在的长度，已用的根数，剩余根数

int dfs(int nowlen,int nowget,int cnt)

{

if(cnt>=n) return 0;

if(get==nowget) return 1;

int i;

int f=0;

if(nowlen==0) f=1;

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

{

if(!used[i])

{

if(len[i]+nowlen==minlen)

{

used[i]=1;

if(dfs(0,nowget+1,nowget))return 1;

used[i] = 0;

return 0;

}

else if(len[i]+nowlen<minlen)

{

used[i]=1;

if(dfs(nowlen+len[i],nowget,i+1))return 1;

used[i]=0;

if(f) return 0;

while(i+1<n&&len[i]==len[i+1]) i++;

}

}

}

return 0;

}

int main()

{

int i, tollen;

while(scanf("%d",&n)!=EOF&&n!=0)

{

tollen = 0;

int j=0,p;

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

{

scanf("%d",&p);

if(p<=50)

{

len[j]=p;

tollen+=len[j];

j++;

}

}

n=j;//总段数

quickSort(len,0,n-1);//将棍子按大到小排列

for(minlen=len[0];;minlen++)

{

if(tollen%minlen) continue;

for(i=0;i<64;i++)

used[i]=0;//重置所有棍子

get=tollen/minlen;

if(dfs(0,0,0))

{

printf("%d\n",minlen);

break;

}

}

}

return 0;

}



#include <stdio.h>

#include <stdlib.h>

int main()

{

int A[100],i=0,j,k,female=0,male=1,x;

for(;;i++)

{

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

if(A[i]==-1)

break;

}

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

{

female=0,male=1;

for(k=1;k<A[j];k++)

{

x=female;

female=male;

male=x+male+1;

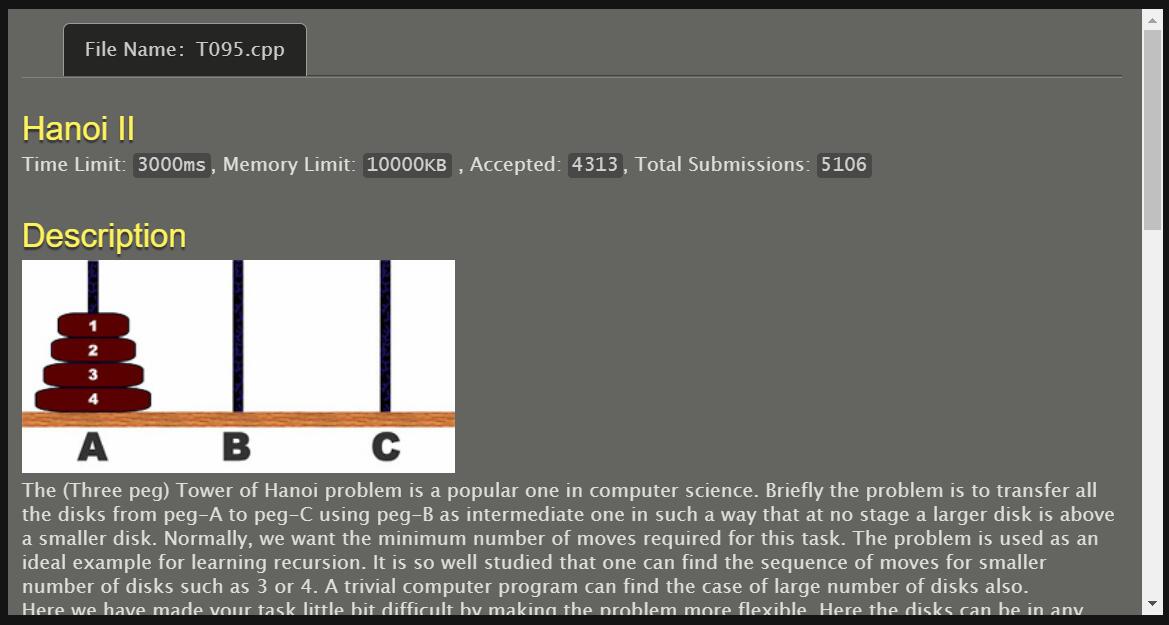
}

printf("%d %d\n",male,female+male+1);

}

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#define M 70

int start[M], targe[M];

long long f(int \*p, int k, int fina)

{

if(k==0) return 0;

if(p[k]==fina) return f(p,k-1,fina);

return f(p,k-1,6-fina-p[k])+(1LL<<(k-1));

}

int main ()

{

long long ans;

int n;

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

{

int i;

for(i=1;i<=n;i++) scanf("%d",&start[i]);

for(i=1;i<=n;i++) scanf("%d",&targe[i]);

int c=n;

for(;c>=1&&start[c]==targe[c];c--);

if(c==0)

{

printf("0\n"); continue;

}

int other=6-start[c]-targe[c];

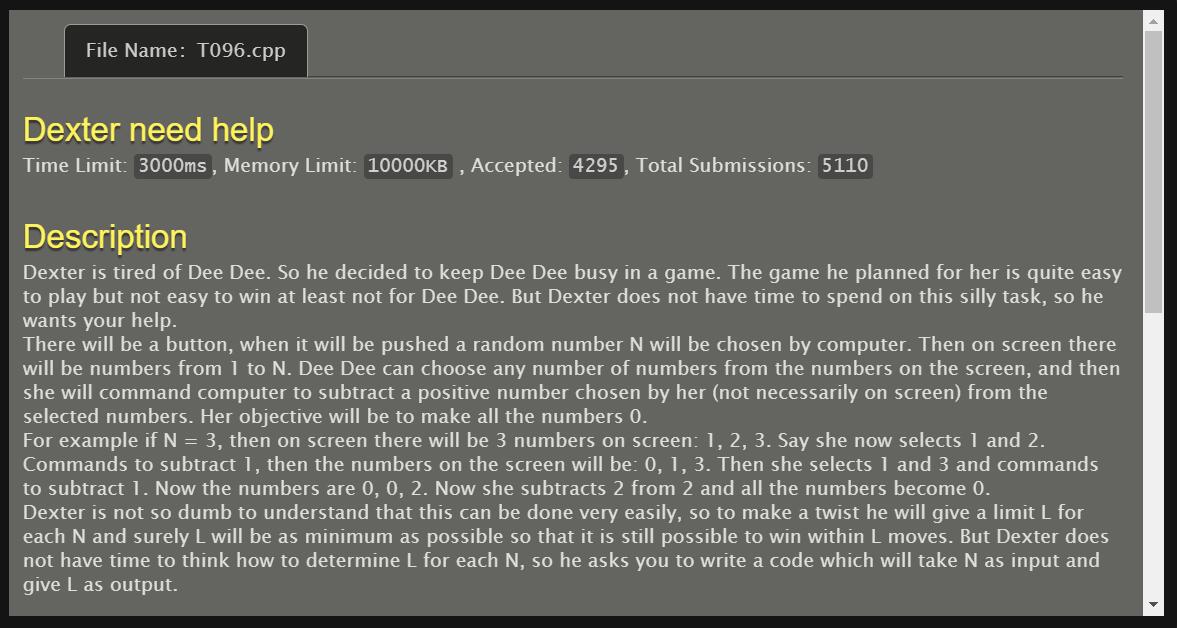
ans=f(start,c-1,other)+f(targe,c-1,other)+1;

printf("%lld\n",ans);

}

return 0;

}



#include <stdio.h>

int fun(int a)

{

if(a==1) return 1;

else return fun(a/2)+1;

}

int main()

{

int a,b[100],i=0,j;

while(1)

{

scanf("%d",&a);

if(a==0)break;

b[i]=fun(a);

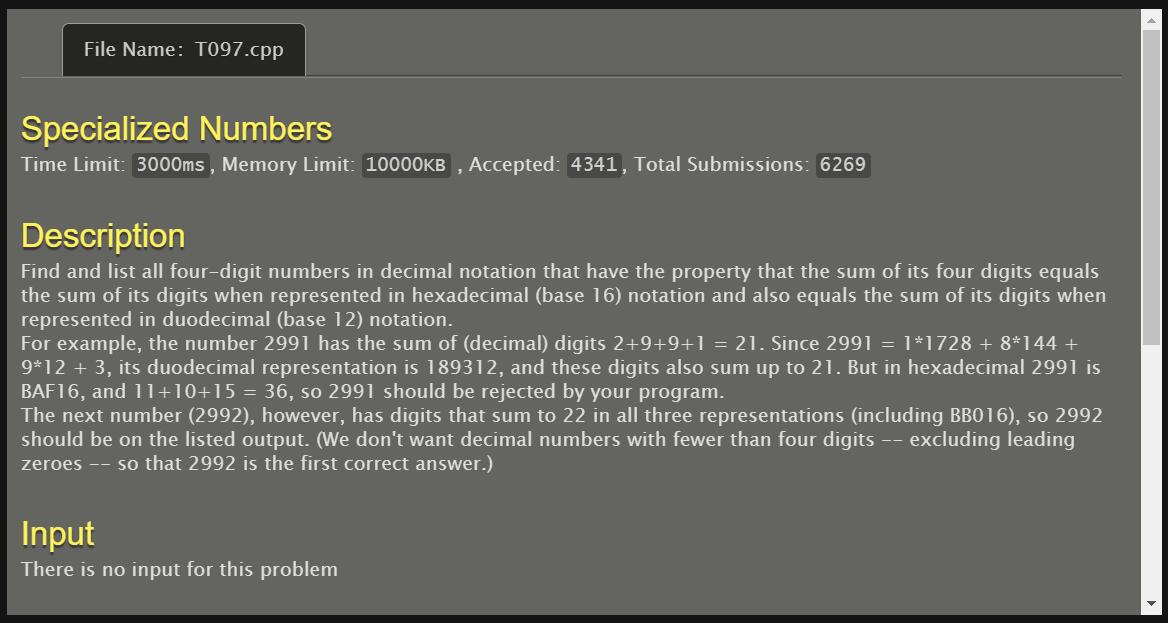
i++;

}

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

return 0;

}



#include <stdio.h>

#include <stdlib.h>

int main()

{

int i,n,sum10,sum12,sum16;

for(i=2992;i<3000;i++){

n=i;

sum10=0;

while(n){

sum10+=n%10;

n/=10;

}

n=i;

sum12=0;

while(n){

sum12+=n%12;

n/=12;

}

n=i;

sum16=0;

while(n){

sum16+=n%16;

n/=16;

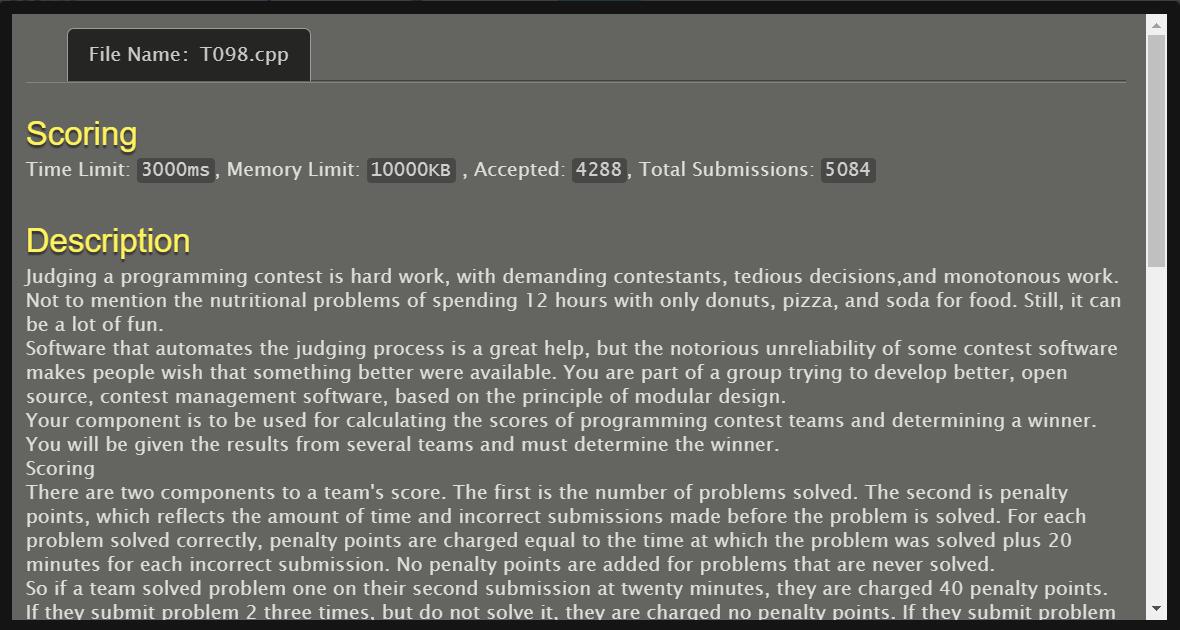
}

if(sum10==sum12&&sum12==sum16) printf("%d\n",i);

}

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

int i,j,sum,min,c,count,n,a,b;

char s1[50],s2[50];

scanf("%d",&n);

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

{

count=sum=0;

scanf("%s",s2);

for(j=0;j<4;j++)

{

scanf("%d%d",&a,&b);

if(b!=0)

{

sum+=(a-1)\*20+b;

count++;

}

}

if(i==0)

{

c=count,min=sum;

strcpy(s1,s2);

}

else if(count>c||(count==c&&sum<min))

{

min=sum;

c=count;

strcpy(s1,s2);

}

}

printf("%s %d %d\n",s1,c,min);

return 0;

}



#include<stdio.h>

int main()

{

int in[20],i,j,cnt=0;

for(i=0;;i++)

{

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

if(in[i]==0)break;

}

for(i=0;in[i]!=0;i++)

for(j=0;in[j]!=0;j++)

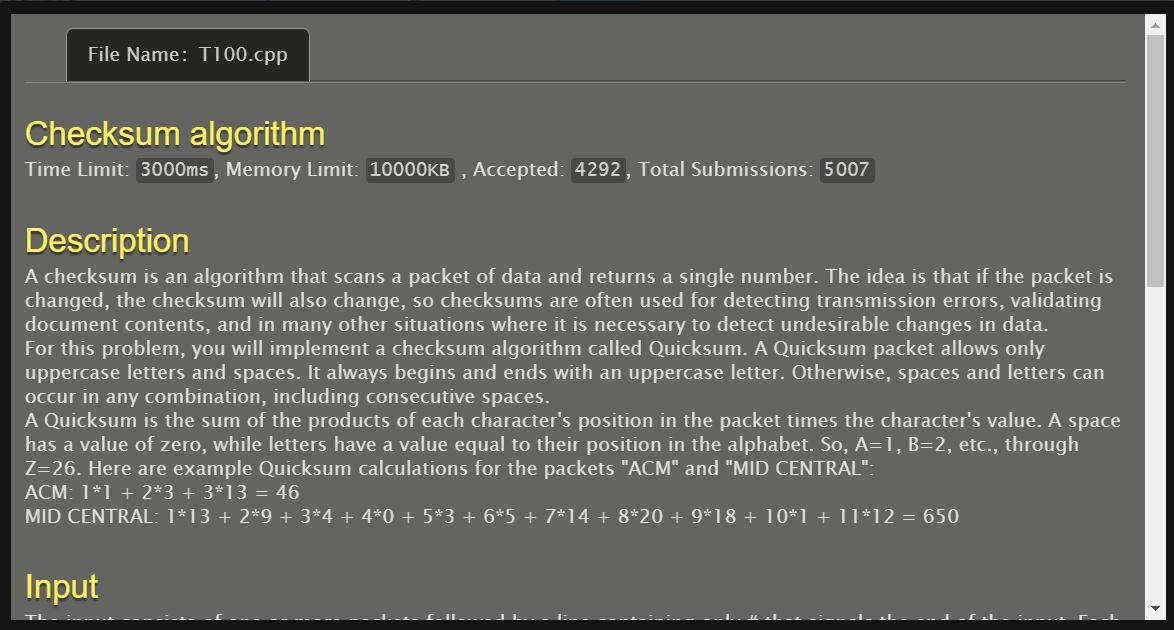
{

if(in[i]/in[j]==2&&in[i]%in[j]==0)cnt++;

}

printf("%d",cnt);

}



#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main()

{

int i,n,t,j;

char s[100][100];

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

gets(s[i]);

if(s[i][0]=='#') break;

}

n=i;

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

t=0;

for(j=0;j<strlen(s[i]);j++)

if(s[i][j]==32) t=t;

else t=t+(j+1)\*(s[i][j]-64);

printf("%d\n",t);

}

return 0;

}