1. Write a program to print the digits of a given number?

#include <stdio.h>

int main()

{

int a,b[20],i=0;

scanf("%d",&a);

while(a){

b[i++]=a%10;

a=a/10;

}

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

printf("%d ",b[j]);

}

return 0;

}

Output:

5013

5 0 1 3

1. Program to write the sum of digits.

#include <stdio.h>

int main()

{

int a,sum=0;

scanf("%d",&a);

while(a){

sum+=a%10;

a=a/10;

}

printf("%d",sum);

return 0;

}

output:

12345

15

1. Find the odd digits sum and even digits sum, and print whether they are equal or not?

#include <stdio.h>

int main()

{

int a,sume=0,sumo=0,b;0

scanf("%d",&a);

while(a){

b=a%10;

if(b%2){

sume+=b;

}

else{

sumo+=b;

}

a=a/10;

}

if(sume==sumo){

printf("Equal");

}

else{

printf("Not Equal");

}

return 0;

}

Output:

222235

Equal

12345

Not equal

1. Check whether the sum of odd positions ,sum of even position are eaual from right ?

#include <stdio.h>

int main()

{

int a,sume=0,sumo=0,i=1;

scanf("%d",&a);

while(a){

if(i%2){

sume+=a%10;

}

else{

sumo+=a%10;

}

i++;

a=a/10;

}

printf("%d",sumo);

printf(" %d ",sume);

if(sume==sumo){

printf("Equal");

}

else{

printf("Not Equal");

}

return 0;

}

1. Print first and last digits of a number?

#include <stdio.h>

int main()

{

int a,n;

scanf("%d",&a);

n=a%10;

while(a/10){

a=a/10;

}

n+=a;

printf("%d",n);

return 0;

}

1. Print the largest and smallest digit ina number

#include <stdio.h>

int main()

{

int a,r,maxi,mini;

scanf("%d",&a);

r=a%10;

maxi=r;

mini=r;

a=a/10;

while(a){

r=a%10;

if(r>maxi){

maxi=r;

}

if(r<mini){

mini=r;

}

a=a/10;

}

printf("%d %d",maxi,mini);

return 0;

}

1. Reverse of a number?

#include <stdio.h>

int main()

{

int a,rev=0;

scanf("%d",&a);

while(a){

rev=rev\*10 +a%10;

a=a/10;

}

printf("%d",rev);

return 0;

}

1. Check whether a number is a Palindrome?

#include <stdio.h>

int main()

{

int n,a,rev=0;

scanf("%d",&a);

n=a;

while(a){

rev=rev\*10 +a%10;

a=a/10;

}

if(rev==n){

printf("Palindrome");

}

else{

printf("Not a palindrome");

}

return 0;

}

1. Print the palindromes in a given range?

#include <stdio.h>

int main()

{

int n,a,rev;

scanf("%d",&n);

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

rev=0;

a=i;

while(a){

rev=rev\*10 +a%10;

a=a/10;

}

if(rev==i){

printf("%d ",i);

}

}

return 0;

}

1. Print the Kth palindrome( for eg: 1st palindrome is 0,2nd palindrome is 1…..15th palindrome is 55….)

#include <stdio.h>

int main()

{

int n,a,rev,count=0;

scanf("%d",&n);

for(int i=0;1;i++){

rev=0;

a=i;

while(a){

rev=rev\*10 +a%10;

a=a/10;

}

if(rev==i){

count++;

if(count==n){

printf("%d",i);

break;

}

}

}

return 0;

}

(OR)

Alternate method:

#include <stdio.h>

int main()

{

int n,a,rev;

scanf("%d",&n);

for(int i=0;n>0;i++){

rev=0;

a=i;

while(a){

rev=rev\*10 +a%10;

a=a/10;

}

if(rev==i){

n--;

}

}

printf("%d",rev);

return 0;

}

1. Check whether a number is automorphic or not. Automorphic number Meaning: sqaure of (6) ends with 6, so 6 is automorphic… 36…sqaure of (25) ends with 25(625)… 11\*11=121 does not ends with 11.

#include <stdio.h>

int main()

{ int n,r;

scanf("%d",&n);

r=n\*n;

while(n){

if(r%10!=n%10){

break;

}

n=n/10;

r=r/10;

}

if(!n){

printf("Automorphic");

}

else{

printf("Not Automorphic");

}

return 0;

}

1. Range of Automorphic numbers.

#include <stdio.h>

int main()

{ int k,r,n;

scanf("%d",&k);

for(int i=0;i<k;i++){

n=i;

r=n\*n;

while(n){

if(r%10!=n%10){

break;

}

n=n/10;

r=r/10;

}

if(!n){

printf("%d ",i);

}

}

return 0;

}

1. Check whether a number is Prime or not.

#include <stdio.h>

#include<math.h>

#include <stdlib.h>

int main()

{

int n,flag=1;

scanf("%d",&n);

if(n==1){

printf("Not Prime");

exit(0);

}

for(int i=2;i<=sqrt(n);i++){

if(n%i==0){

flag=0;

break;

}

}

if(flag){

printf("Prime");

}

else{

printf("Not Prime");

}

return 0;

}

1. Prime numbers range.

#include <stdio.h>

#include<math.h>

int main()

{

int n,flag=1;

scanf("%d",&n);

for(int j=2;j<n;j++){

flag=1;

for(int i=2;i<=sqrt(j);i++){

if(j%i==0){

flag=0;

}

}

if(flag){

printf("%d ",j);

}

}

return 0;

}

1. Printing prime nmbers from r1 to r2.

#include <stdio.h>

#include<math.h>

int main()

{

int n,flag,r1,r2;

scanf("%d %d",&r1,&r2);

for(int j=r1;j<=r2;j++){

flag=1;

for(int i=2;i<=sqrt(j);i++){

if(j%i==0){

flag=0;

}

}

if(flag){

printf("%d ",j);

}

}

return 0;

}

1. Sum of prime numbers in a given range.

#include <stdio.h>

#include<math.h>

int main()

{

int n,flag,r1,r2,sum=0;

scanf("%d %d",&r1,&r2);

for(int j=r1;j<=r2;j++){

flag=1;

for(int i=2;i<=sqrt(j);i++){

if(j%i==0){

flag=0;

}

}

if(flag){

**sum+=j;**

}

}

printf("%d",sum);

return 0;

}

1. Print Kth prime Number.

#include <stdio.h>

#include<math.h>

int main()

{

int n,count=0;

scanf("%d",&n);

for(int j=2;;j++){

int flag=1;

for(int i=2;i<=sqrt(j);i++){

if(j%i==0){

flag=0;

break;

}

}

if(flag){

count++;

}

if(count==n){

printf("%d",j);

break;

}

}

return 0;

}

1. Fibonacci series.

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n];

arr[0]=0;

arr[1]=1;

for(int i=2;i<n;i++){

arr[i]=arr[i-1]+arr[i-2];

}

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

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

}

}

(OR)

#include <stdio.h>

int main()

{

int n,f1=-1,f2=1,f3;

scanf("%d",&n);

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

f3=f1+f2;

f1=f2;

f2=f3;

printf("%d ",f3);

}

}

1. Nth fibonacci number.

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n];

arr[0]=0;

arr[1]=1;

for(int i=2;i<n;i++){

arr[i]=arr[i-1]+arr[i-2];

}

printf("%d",arr[n-1]);

}

1. Print the no. Of digits in nth fibonacci?

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n];

arr[0]=0;

arr[1]=1;

for(int i=2;i<n;i++){

arr[i]=arr[i-1]+arr[i-2];

}

int count=0;

printf("%d ",arr[n-1]);

do{

arr[n-1]=arr[n-1]/10;

count++;

}while(arr[n-1]);

printf("%d",count);

}

(OR)

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n];

arr[0]=0;

arr[1]=1;

for(int i=2;i<n;i++){

arr[i]=arr[i-1]+arr[i-2];

}

int c=printf("%d",arr[n-1]);

printf(" %d",c);

}

Note: printf() return the number of characters in argument.

Printf(“hello”)//5

C=1234;

Printf(“%d”,c)//4

PATTERNS:

Input;5

Output:

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

Output2:

A

A B

A B C

A B C D

A B C D E

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

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

for(int j=1;j<=i;j++){

printf("%d ",j);// printf(“%c”,j+64);to print second output.

}

printf("\n");

}

}

Note: Add the below code inside second loop to get another pattern.

if(j<i){

printf("\*");

}

Pattern:

1

1\*2

1\*2\*3

1\*2\*3\*4

1\*2\*3\*4\*5

22.

5

5 4

5 4 3

5 4 3 2

5 4 3 2 1

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

for(int i=5;i>=1;i--){

for(int j=5;j>=i;j--){

printf("%d ",j);

}

printf("\n");

}

}

23.

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

for(int i=5;i>=1;i--){

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

}

24.

1

2 1

3 2 1

4 3 2 1

5 4 3 2 1

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

for(int i=1;i<=5;i++){

for(int j=i;j>=1;j--){

printf("%d ",j);

}

printf("\n");

}

}

25.

5

4 5

3 4 5

2 3 4 5

1 2 3 4 5

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

for(int i=n;i>=1;i--){

for(int j=i;j<=n;j++){

printf("%d ",j);

}

printf("\n");

}

}

26.

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

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

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

if(i==1||i==n||j==1||j==n){

printf("\*");

}

else{

printf(" ");

}

}

printf("\n");

}

}

27.

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1

int main()

{

for(int i=1;i<=5;i++){

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

for(int i=5;i>=1;i--){

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

}

Note: If you don’t want the 12345 two times, you can start the second loop from i=4.

28.

1

12

123

1234

12345

#include <stdio.h>

int main()

{

for(int i=1;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d",j);

}

printf("\n");

}

}

29.

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

int main()

{

for(int i=1;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

}

30.

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1

#include <stdio.h>

int main()

{

for(int i=1;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

for(int i=4;i>=1;i--){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d ",j);

}

printf("\n");

}

}



1

121

12321

1234321

123454321

#include <stdio.h>

int main()

{

for(int i=1;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d",j);

}

for(int j=i-1;j>=1;j--){

printf("%d",j);

}

printf("\n");

}

}

32.

1

121

12321

1234321

123454321

1234321

12321

121

1

#include <stdio.h>

int main()

{

for(int i=1;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d",j);

}

for(int j=i-1;j>=1;j--){

printf("%d",j);

}

printf("\n");

}

for(int i=4;i>=1;i--){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%d",j);

}

for(int j=i-1;j>=1;j--){

printf("%d",j);

}

printf("\n");

}

}

33.

0

101

21012

3210123

432101234

54321012345

#include <stdio.h>

int main()

{

for(int i=0;i<=5;i++){

for(int k=5;k>i;k--){

printf(" ");

}

for(int j=i;j>=0;j--){

printf("%d",j);

}

for(int j=1;j<=i;j++){

printf("%d",j);

}

printf("\n");

}

}

34.

ABCDEFGGFEDCBA

ABCDEFFEDCBA

ABCDEEDCBA

ABCDDCBA

ABCCBA

ABBA

AA

#include <stdio.h>

int main()

{

for(int i=7;i>=1;i--){

for(int k=7;k>i;k--){

printf(" ");

}

for(int j=1;j<=i;j++){

printf("%c",j+64);

}

for(int j=i;j>=1;j--){

printf("%c",j+64);

}

printf("\n");

}

}

35.

1234\*

123\*5

12\*45

1\*345

\*2345

#include <stdio.h>

int main()

{

for(int i=1;i<=5;i++){

for(int j=1;j<=5;j++){

if(j==(5-i+1)){

printf("\*");

}

else{

printf("%d",j);

}

}

printf("\n");

}

}

36.

4444444

4333334

4322234

4321234

4322234

4333334

4444444

#include <stdio.h>

int main()

{

for(int i=4;i>=1;i--){

for(int j=4;j>=1;j--){

if(i>=j){

printf("%d",i);

}

else{

printf("%d",j);

}

}

for(int j=2;j<=4;j++){

if(i>=j){

printf("%d",i);

}

else{

printf("%d",j);

}

}

printf("\n");

}

for(int i=2;i<=4;i++){

for(int j=4;j>=1;j--){

if(i>=j){

printf("%d",i);

}

else{

printf("%d",j);

}

}

for(int j=2;j<=4;j++){

if(i>=j){

printf("%d",i);

}

else{

printf("%d",j);

}

}

printf("\n");

}

}

37.

1234

2341

3421

4321

#include <stdio.h>

int main()

{

for(int i=1;i<=4;i++){

for(int j=i;j<=4;j++){

printf("%d",j);

}

for(int j=i-1;j>=1;j--){

printf("%d",j);

}

printf("\n");

}

}

Note: If you want the code to work for any number, scan the input(n),replace 4 with n. By doing this we can get the pattern for any number n.

38.

1

2 3

4 5 6

7 8 9 10

#include <stdio.h>

int main()

{ int c=0;

for(int i=1;i<=4;i++){

for(int j=1;j<=i;j++){

printf("%d ",++c);

}

printf("\n");

}

}

39.

5

55555

45555

34555

23455

12345

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

for(int i=n;i>=1;i--){

for(int j=i;j<=n;j++){

printf("%d",j);

}

for(int j=i-1;j>=1;j--){

printf("%d",n);

}

printf("\n");

}

}

40.

1 1

2 3 3 2

4 5 6 6 5 4

7 8 9 10 10 9 8 7

#include <stdio.h>

int main()

{ int c=0,e;

for(int i=1;i<=4;i++){

e=c+1;

for(int j=1;j<=i;j++){

printf("%d ",++c);

}

int d=c;

for(int j=d;j>=e;j--){

printf("%d ",j);

}

printf("\n");

}

}

41.

1

10

101

1010

#include <stdio.h>

int main()

{

for(int i=1;i<=4;i++){

for(int j=1;j<=i;j++){

printf("%d",j%2);

}

printf("\n");

}

}

**ARRAYS:**

The remainder when divided by 9 is the sum of the digits of the nuber till we get single digit number.

Eg: 1234%9=1 =>1+2+3+4=10 =>1+0=1

12345%9=6 => 1+2+3+4+5=15 => 1+5 =6

If the number is divisble by 9 then we will print 9;  
27%9==0 2+7=9

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n],sum=0;

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

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

sum+=arr[i];

}

if(sum%9==0){

printf("9");

}

else{

printf("%d",sum%9);

}

return 0;

}

If we don’t use the above code, it is a long process as below:

int main()

{

int n;

scanf("%d",&n);

int arr[n],sum=0;

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

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

sum+=arr[i];

}

int a=sum;

while(a/10){

a=0;

while(sum/10){

a+=sum%10;

sum=sum/10;

}

a+=sum;

sum=a;

}

printf("%d",a);

return 0;

}

2.

Sum of even,sum of odd, sum of prime.

#include <stdio.h>

#include <math.h>

int main()

{

int n,sumo=0,sume=0;

scanf("%d",&n);

int a[n];

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

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

if(a[i]%2){

sumo+=a[i];

}

else{

sume+=a[i];

}

}

int count,sum=0;

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

count=0;

for(int j=2;j<=sqrt(a[i]);j++)

{

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

count++;

break;

}

}

if(!count){

sum+=a[i];

}

}

printf("Sum of prime numbers:%d\n",sum);

printf("sum of even numbers:%d\n",sume);

printf("sum of odd numbers:%d\n",sumo);

}

1. Storing the even numbers in one array and odd numbers in another array.

#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int arr[n],b[n],c[n],p=0,q=0;

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

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

}

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

if(arr[i]%2==0){

b[p++]=arr[i];

}

else{

c[q++]=arr[i];

}

}

for(int i=0;i<p;i++){

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

}

printf("\n");

for(int i=0;i<q;i++){

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

}

return 0;

}

1. Reversing the elements of an array.

#include <stdio.h>

int main()

{

int n,temp;

scanf("%d",&n);

int a[n];

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

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

}

for(int i=0;i<n/2;i++){

temp=a[i];

a[i]=a[n-i-1];

a[n-i-1]=temp;

}

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

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

}

}

1. Reversing the first half of the elements.

//reversing first half

#include <stdio.h>

int main()

{

int n,temp;

scanf("%d",&n);

int a[n];

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

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

}

for(int i=0,j=n/2 -1;i<j;i++,j--){

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

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

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

}

}

1. Reversing the second half of the array.

#include <stdio.h>

int main()

{

int n,temp;

scanf("%d",&n);

int a[n];

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

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

}

for(int i=n/2,j=n-1;i<j;i++,j--){

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

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

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

}

}

1. Find the first two largest numbers.

#include<stdio.h>

void main()

{

int n,i,max1,max2;

scanf("%d",&n);

int a[n];

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

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

max1=max2=a[0];

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

{

if(a[i]>max1)

{

max2=max1;

max1=a[i];

}

else if(a[i]>=max2&&a[i]!=max1)

max2=a[i];

else if(max1==max2)

max2=a[i];

}

printf("%d %d",max1,max2);

}

NOTE: WRITE < IN PLACE OF > IF U WANT TO GET TWO SMALLEST NUMBERS.

1. Write one fibonacci and one prime and so on…. for n numbers

N=10

0 2 1 3 1 5 2 7 3 11

#include<stdio.h>

#include<math.h>

void main()

{

int n,k=2,flag;

scanf("%d",&n);

int arr[n];

arr[0]=0;

arr[2]=1;

for(int i=4;i<n;i++){

arr[i]=arr[i-2]+arr[i-4];

}

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

flag=1;

for(int j=2;j<=k/2;j++){

if(k%j==0){

flag=0;

break;

}

}

if(flag){

arr[i]=k;

i+=2;

}

k++;

}

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

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

}

}

1. rotate the elements

#include<stdio.h>

#include<math.h>

void main()

{

int n;

scanf("%d",&n);

int a[n],i;

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

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

}

int c=a[n-1];

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

a[i]=a[i-1];

}

a[i]=c;

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

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

}

}

1. K rotations

#include<stdio.h>

#include<math.h>

void main()

{

int n,m;

scanf("%d",&n);

int a[n],i;

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

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

}

scanf("%d",&m);

for(int j=0;j<m;j++){

int c=a[n-1];

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

a[i]=a[i-1];

}

a[i]=c;

}

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

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

}

}

(OR)

#include<stdio.h>

#include<math.h>

void main()

{

int n,m;

scanf("%d",&n);

int a[n],arr[n],i;

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

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

}

scanf("%d",&m);

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

arr[i]=a[(i+m)%n];

}

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

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

}

}

2-D ARRAYS

1. Scanning and printing a 2-d array.

#include <stdio.h>

int main()

{ int m,n;

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

int a[m][n];

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

for(int j=0;j<n;j++){

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

}

}

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

for(int j=0;j<n;j++){

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

}

printf("\n");

}

return 0;

}

1. Sum of the diagonal elements in array.

#include <stdio.h>

int main()

{ int sum=0,n;

scanf("%d",&n);

int a[n][n];

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

for(int j=0;j<n;j++){

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

if(i==j || i==n-j-1){

sum+=a[i][j];

}

}

}

printf("%d",sum);

return 0;

}

1. Sum of boundary elements

#include <stdio.h>

int main()

{

int m,n,sum=0;

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

int a[m][n],b[n][m];

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

for(int j=0;j<n;j++){

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

if(i==0||j==0||i==m-1||j==n-1){

sum+=a[i][j];

}

}

}

printf("%d",sum);

return 0;

}

1. Left rotation of a matrix.

#include <stdio.h>

int main()

{

int m,n;

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

int a[m][n],b[n][m];

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

for(int j=0;j<n;j++){

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

}

}

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

for(int j=0;j<n;j++){

b[n-1-j][i]=a[i][j];

}

}

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

for(int j=0;j<m;j++){

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

}

printf("\n");

}

return 0;

}

(OR)

#include <stdio.h>

int main()

{

int m,n;

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

int a[m][n],b[n][m];

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

for(int j=0;j<n;j++){

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

}

}

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

for(int j=0;j<m;j++){

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

}

printf("\n");

}

return 0;

}

Rotating right.

#include <stdio.h>

int main()

{

int m,n;

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

int a[m][n],b[n][m];

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

for(int j=0;j<n;j++){

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

}

}

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

for(int j=0;j<n;j++){

b[j][m-1-i]=a[i][j];

}

printf("\n");

}

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

for(int j=0;j<m;j++){

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

}

printf("\n");

}

return 0;

}

FUNCTIONS:

#include <stdio.h>

int pal(int n){

int temp=n,rev=0;

while(temp){

rev=rev\*10+temp%10;

temp=temp/10;

}

if(n==rev){

return 1;

}

return 0;

}

int main()

{

int n,flag=1;

scanf("%d",&n);

int a[n];

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

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

}

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

if(!pal(a[i])){

flag=0;

break;

}

}

if(flag){

printf("true");

}

else{

printf("false");

}

return 0;

}

Binary palindrome.

#include <stdio.h>

int binary(int n){

int arr[20],i=0;

while(n){

arr[i++]=n%2;

n=n/2;

}

for(int j=0;j<i/2;j++){

if(arr[j]!=arr[i-1-j]){

return 0;

}

}

return 1;

}

int main()

{

int n,flag=1;

scanf("%d",&n);

int a[n];

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

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

}

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

if(!binary(a[i])){

flag=0;

break;

}

}

if(flag){

printf("true");

}

else{

printf("false");

}

return 0;

}

RECCURSION:

1. FACTORIAL

#include <stdio.h>

int fact(int n){

if(n<=1){

return 1;

}

return n\*fact(n-1);

}

int main()

{

printf("%d",fact(5));

}

1. SUM OF NATURAL NUMBERS.

#include <stdio.h>

int sumofn(int n){

if(n<=1){

return 1;

}

return n+sumofn(n-1);

}

int main()

{

printf("%d",sumofn(5));

}

1. SUM OF DIGITS OF NUMBER.

#include <stdio.h>

int sumofdigits(int n){

if(n==0){

return 0;

}

return n%10+sumofdigits(n/10);

}

int main()

{

printf("%d",sumofdigits(54321));

}

1. REVERSE OF A NUMBER.

#include <stdio.h>

int rev(int n){

static int r=0;

if(n==0){

return 0;

}

r=r\*10+n%10;

rev(n/10);

return r;

}

int main()

{

printf("%d",rev(123));

}

1. MULTIPLICATION OF TWO NUMBERS.

#include <stdio.h>

int mul(int n,int m){

if(!n || !m){

return 0;

}

return n+mul(n,m-1);

}

int main()

{

printf("%d",mul(1,50));

}

1. Floor of log a base b using recursion.

#include <stdio.h>

int loger(int a,int b){

if(a<b){

return 0;

}

return 1+loger(a/b,b);

}

int main()

{

printf("%d",loger(33,2));

}

POINTERS:

#include <stdio.h>

#include<stdlib.h>

int main()

{

int a[5][5]={{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1}};

int b[5][5]={{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1},{1,2,1,2,1}};

int c[5][5]={0};

for(int i=0;i<5;i++){

for(int j=0;j<5;j++){

for(int k=0;k<5;k++){

c[i][j]+=a[i][k]\*b[k][j];

}

}

}

for(int i=0;i<5;i++){

for(int j=0;j<5;j++){

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

}

printf("\n");

}

int sum=0;

for(int i=0;i<5;i++){

if(i%2==0){

for(int j=0;j<=5/2;j++){

sum+=c[j][i];

}

}

else{

for(int j=5/2 +1;j<5;j++){

sum+=c[j][i];

}

}

}

for(int i=0;i<5;i++){

if(i%2==0){

for(int j=0;j<=5/2;j++){

sum+=c[i][j];

}

}

else{

for(int j=5/2 +1;j<5;j++){

sum+=c[i][j];

}

}

}

printf("%d",sum);

}

STRINGS:

gets(a)

Input:hello world

Printf(“%s”,a)

hello world

Scanf(“%d”,a)

Input:hello world

Printf(“%s”,a)

hello

Strcpy()-used to copy

Strncpy()-used to copy n characters.

Strcat()- concatenation

Strcmp()-compare the strings

Strcmp() will return 0 if the strings are equal.

It prints negative value when:

“cse”,”ece” first char in first string is not matching with first char in second string.

c-e= (99-101)= -2.

Similar is the case with positive value.

“hello hi”

“hello bb”

Output: h-b=6

Strlwr() and strupr() are used to convert strings to lower and upper case respectively.

1. Find the length of string without using buit-in function.

#include <stdio.h>

#include<string.h>

int main()

{

char a[100]="cse gurus";

int count=0;

for(int i=0;a[i]!='\0';i++){

count++;

}

printf("%d",count);

return 0;

}

1. Copy the string without using buit-in function.

#include <stdio.h>

#include<string.h>

int main()

{

char a[100]="cse gurus";

char b[100],i;

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

b[i]=a[i];

}

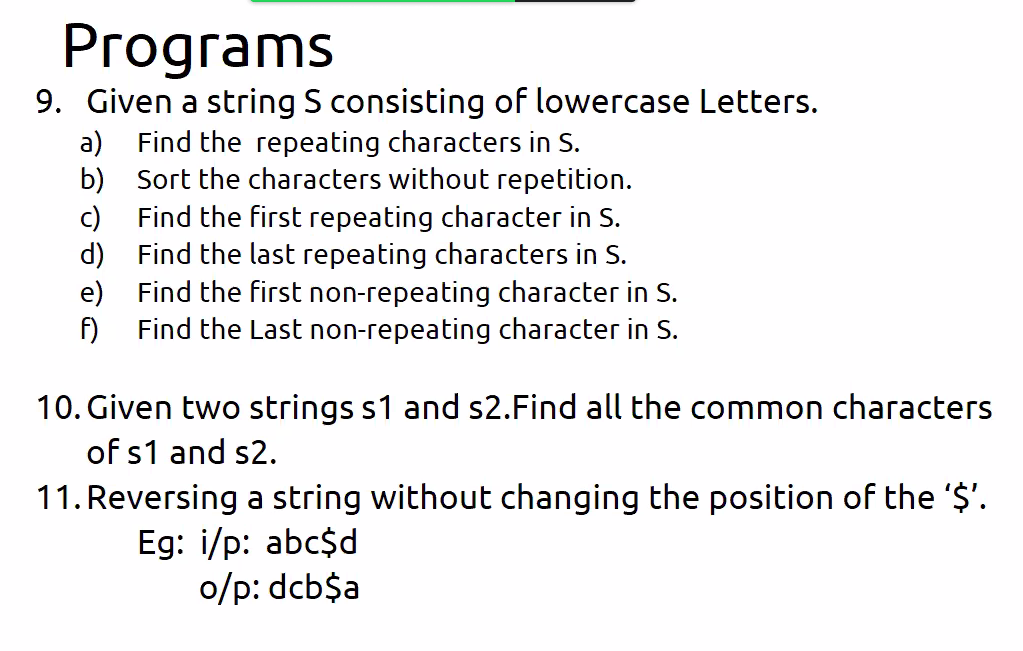
b[i]='\0';

printf("%s",a);

printf("%s",b);

return 0;

}



9.

a.

#include <stdio.h>

int main()

{

char a[100]="cse gurus";

int b[25]={0};

for(int i=0;a[i]!='\0';i++){

b[a[i]-'a']++;

}

for(int i=0;i<26;i++){

if(b[i]!=0){

printf("%c-%d\n",i+97,b[i]);

}

}

return 0;

} (OR)

#include <stdio.h>

int main()

{

char a[100];

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

for(int i=0;i<123;i++){

if(b[i]>1)

printf("%c-%d\n",i,b[i]);

}

}

(OR)

#include <stdio.h>

int main()

{

char a[100];

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

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

if(b[a[i]]>1){

printf("%c-%d\n",a[i],b[a[i]]);

b[a[i]]=0;

}

}

}

C.

#include <stdio.h>

int main()

{

char a[100];

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

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

if(b[a[i]]>1){

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

break;

}

}

}

D.

#include <stdio.h>

int main()

{

char a[100],t;

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

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

if(b[a[i]]>1){

t=a[i];

b[a[i]]=0;

}

}

printf("%c",t);

}

E.

#include <stdio.h>

int main()

{

char a[100];

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

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

if(b[a[i]]==1){

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

break;

}

}

}

F.

#include <stdio.h>

int main()

{

char a[100],t='\0';

scanf("%s",a);

int b[123]={0},n;

for(n=0;a[n]!='\0';n++);

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

b[a[i]]++;

}

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

if(b[a[i]]==1){

t=a[i];

b[a[i]]=0;

}

}

printf("%c",t);

}

11.

#include <stdio.h>

#include<string.h>

void reverse(char\* s){

int n=strlen(s);

char temp;

for(int i=0,j=n-1;i<j;){

if(s[i]=='$'){

i++;

}

else if(s[j]=='$'){

j--;

}

else{

temp=s[i];

s[i]=s[j];

s[j]=temp;

i++;

j--;

}

}

printf("%s",s);

}

int main()

{

char str[100];

scanf("%s",str);

reverse(str);

return 0;

}

1. Remove the consecutive repeated characters in a string.

Eg: freeze-> output freze

#include <stdio.h>

#include<string.h>

void rcrc(char\* s){

int n=strlen(s);

int j=0;

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

if(s[i]!=s[i-1]){

s[j++]=s[i-1];

}

}

s[j++]=s[n-1];

s[j]='\0';

printf("%s",s);

}

int main()

{

char str[100];

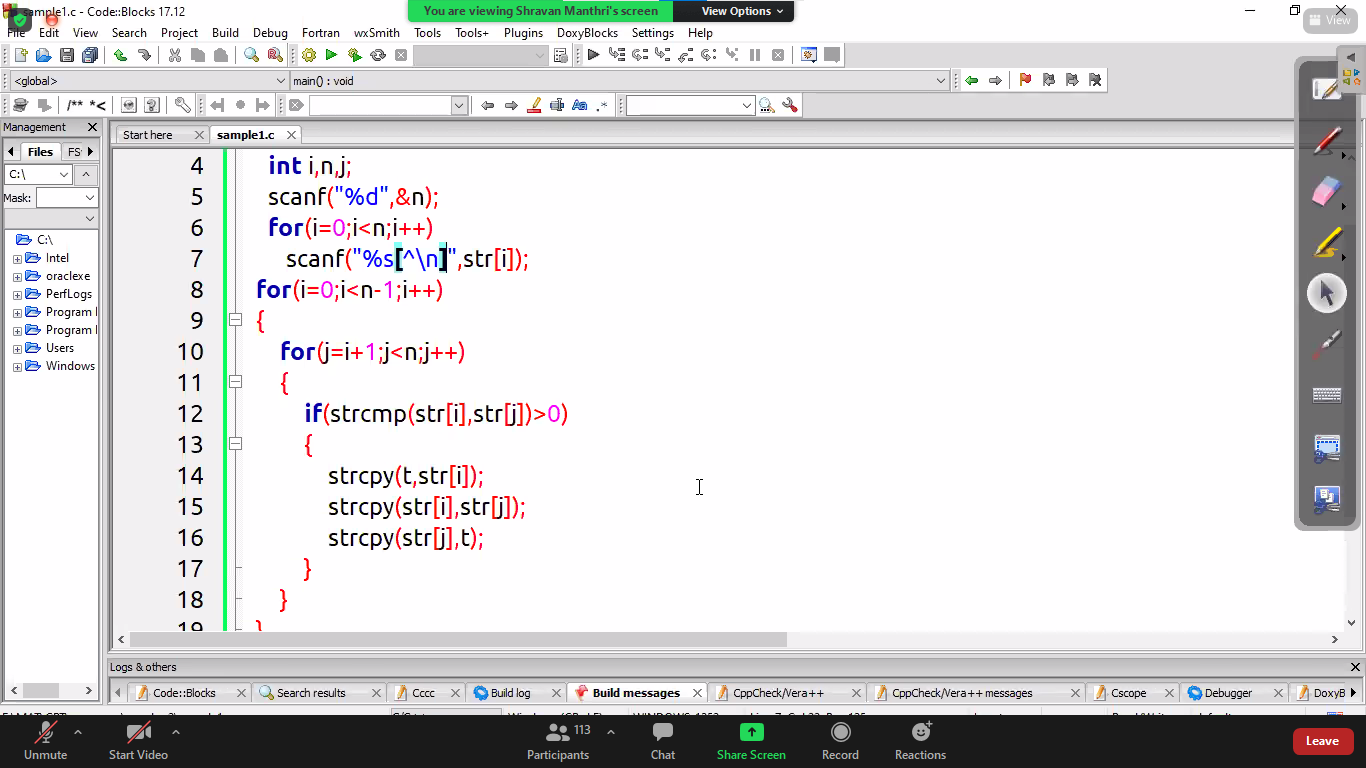
scanf("%s",str);

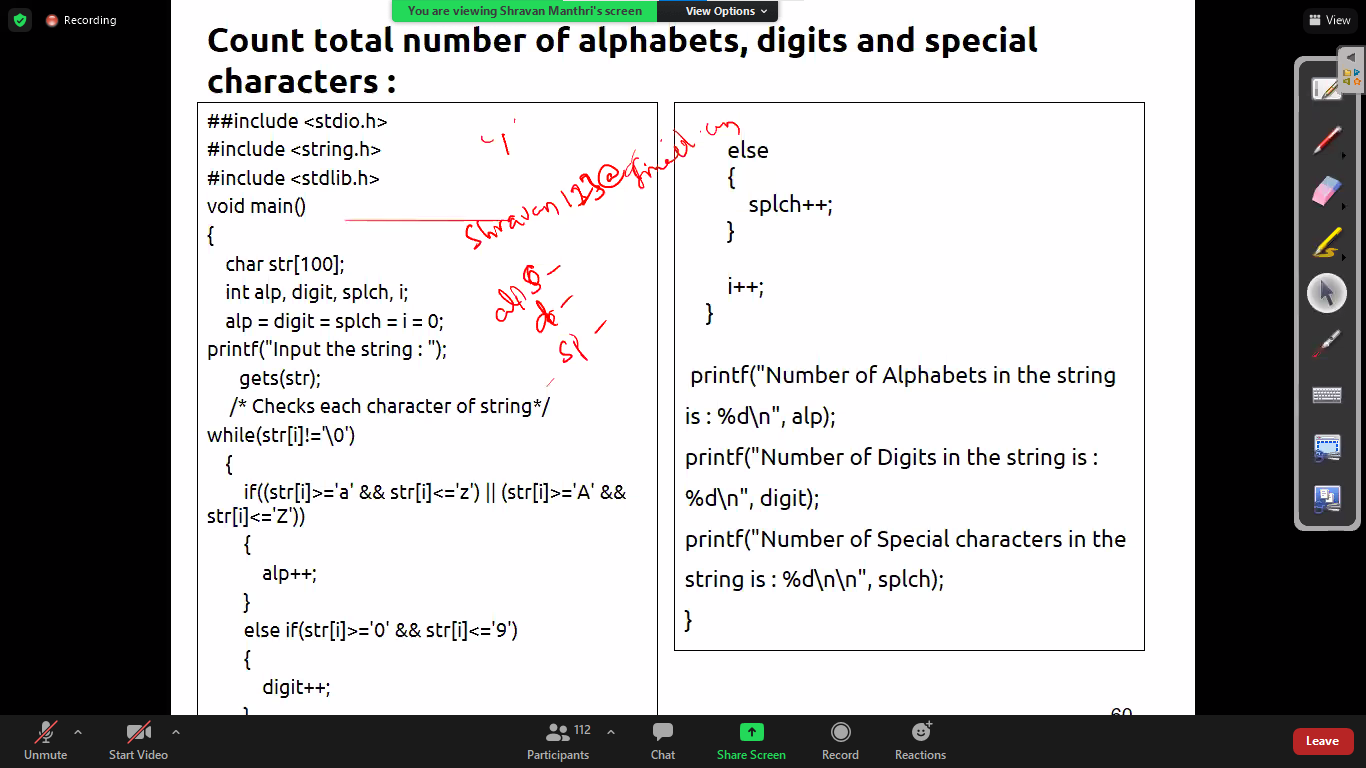
rcrc(str);

return 0;

}

Dictionary





Print the sum of last four digits of a number plate.

#include <stdio.h>

#include<string.h>

int main()

{

char str[100];

scanf("%s",str);

int n=strlen(str),sum=0;

for(int i=n-4;i<n;i++){

sum+=str[i]-48;

}

printf("%d",sum);

return 0;

}

void main()

{

char str[100],s2[100];

int sum=0,i,n,j=0,k;

gets(str);

for(n=0;str[n]!='\0';n++);

for(i=n-4;i<n;i++)

s2[j++]=str[i];

k=atoi(s2);

while(k!=0)

{

sum=sum+k%10;

k=k/10;

}

printf("%d",sum);

}

Finding the common characters in two strings

#include <stdio.h>

#include<string.h>

void commonchar(char \*s1,char\* s2){

int c1[26]={0},c2[26]={0};

for(int i=0;i<strlen(s1);i++){

c1[s1[i]-'a']++;

}

for(int i=0;i<strlen(s2);i++){

c2[s2[i]-'a']++;

}

for(int i=0;i<strlen(s1);i++){

if(c1[s1[i]]>0 && c2[s1[i]]>0){

printf("%c ",s1[i]);

c1[s1[i]]=0;

}

}

}

int main()

{

char str1[100],str2[100];

scanf("%s",str1);

scanf("%s",str2);

commonchar(str1,str2);

return 0;

}

**#include <stdio.h>**

**#include<string.h>**

**int main()**

**{**

**int c1[26]={0},c2[26]={0},c3[10]={0},j;**

**char str[50],p;**

**scanf("%s",str);**

**for(int i=0;i<strlen(str);i++){**

**if(str[i]>='A'&& str[i]<='Z'){**

**if(c1[str[i]-'A']==0 && c2[str[i]-'A']==0 ){**

**c1[str[i]-'A']++;**

**c2[str[i]-'A']++;**

**}**

**else{**

**char p=str[i];**

**while(c1[p-'A']!=0 && c2[p-'A']!=0 ){**

**if(p=='Z'){**

**p='A';**

**}**

**else{**

**p++;**

**}**

**}**

**str[i]=p;**

**c1[p-'a']++;**

**c2[p-'a']++;**

**}**

**}**

**else if(str[i]>='a'&& str[i]<='z'){**

**if(c1[str[i]-'a']==0 && c2[str[i]-'a']==0 ){**

**c1[str[i]-'a']++;**

**c2[str[i]-'a']++;**

**}**

**else{**

**char p=str[i];**

**while(c1[p-'a']!=0 && c2[p-'a']!=0 ){**

**if(p=='z'){**

**p='a';**

**}**

**else{**

**p++;**

**}**

**}**

**str[i]=p;**

**c1[p-'a']++;**

**c2[p-'a']++;**

**}**

**}**

**else if(str[i]>='0'&& str[i]<='9'){**

**if(c3[str[i]-'0']==0){**

**c3[str[i]-'0']++;**

**}**

**else{**

**p=str[i];**

**while(c3[p-'0']!=0){**

**if(p=='9'){**

**p='0';**

**}**

**else{**

**p++;**

**}**

**}**

**str[i]=p;**

**c3[p-'0']++;**

**}**

**}**

**}**

**printf("%s",str);**

**return 0;**

**}**

**(OR)**

**void main()**

**{**

**char str[100];**

**int i,n,c[123]={0};**

**gets(str);**

**for(n=0;str[n]!='\0';n++);**

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

**{**

**if(c[str[i]]!=0)**

**{**

**if(str[i]>='a' &&str[i]<='z')**

**{**

**str[i]=(str[i]-96)%26+'a';**

**}**

**else if(str[i]>='0'&&str[i]<='9')**

**str[i]=(str[i]-47)%10+'0';**

**else**

**str[i]=(str[i]-64)%26+'A';**

**}**

**else**

**{**

**c[str[i]]++;**

**if(str[i]>='a'&&str[i]<='z')**

**c[str[i]-32]++;**

**else if(str[i]>='A'&&str[i]<='Z')**

**c[str[i]+32]++;**

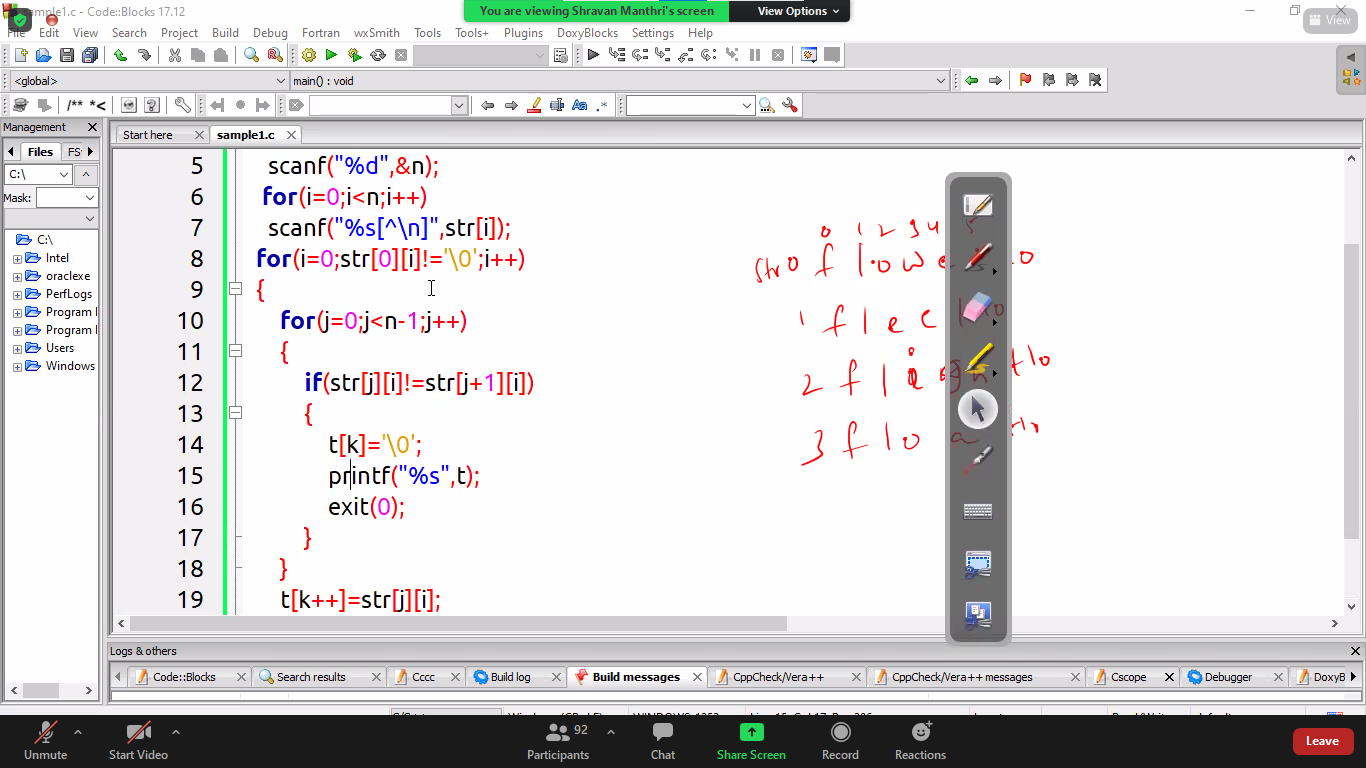
**i++;**

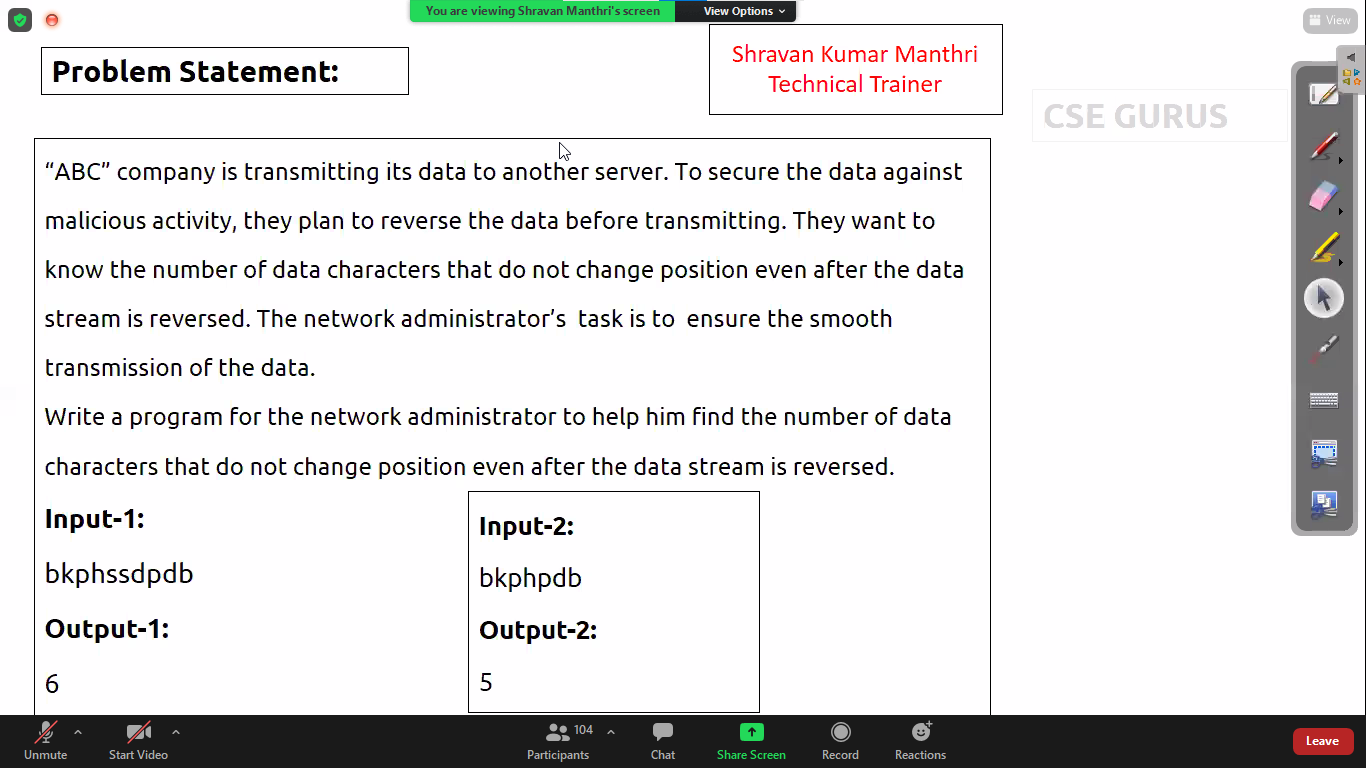
**}**

**}**

**printf("%s",str);**

**}**





#include <stdio.h>

int main()

{

char s[100];

int c=0,n;

scanf("%s",s);

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

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

if(s[i]==s[j])

c+=2;

}

if(n%2)

c+=1

printf("%d",c);

}

STRUCTURES:

#include <stdio.h>

#include<string.h>

struct college{

char name[100];

char city[100];

int year;

float percentage;

};

int main()

{

int n;

scanf("%d",&n);

struct college a[n];

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

scanf("%s",a[i].name);

scanf("%s",a[i].city);

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

scanf("%f",&a[i].percentage);

}

struct college temp;

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

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

if(strcmp(a[i].name,a[j].name)>0){

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

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

printf("The Details of college %d is:\n",i+1);

printf("Name:%s\n",a[i].name);

printf("city:%s\n",a[i].city);

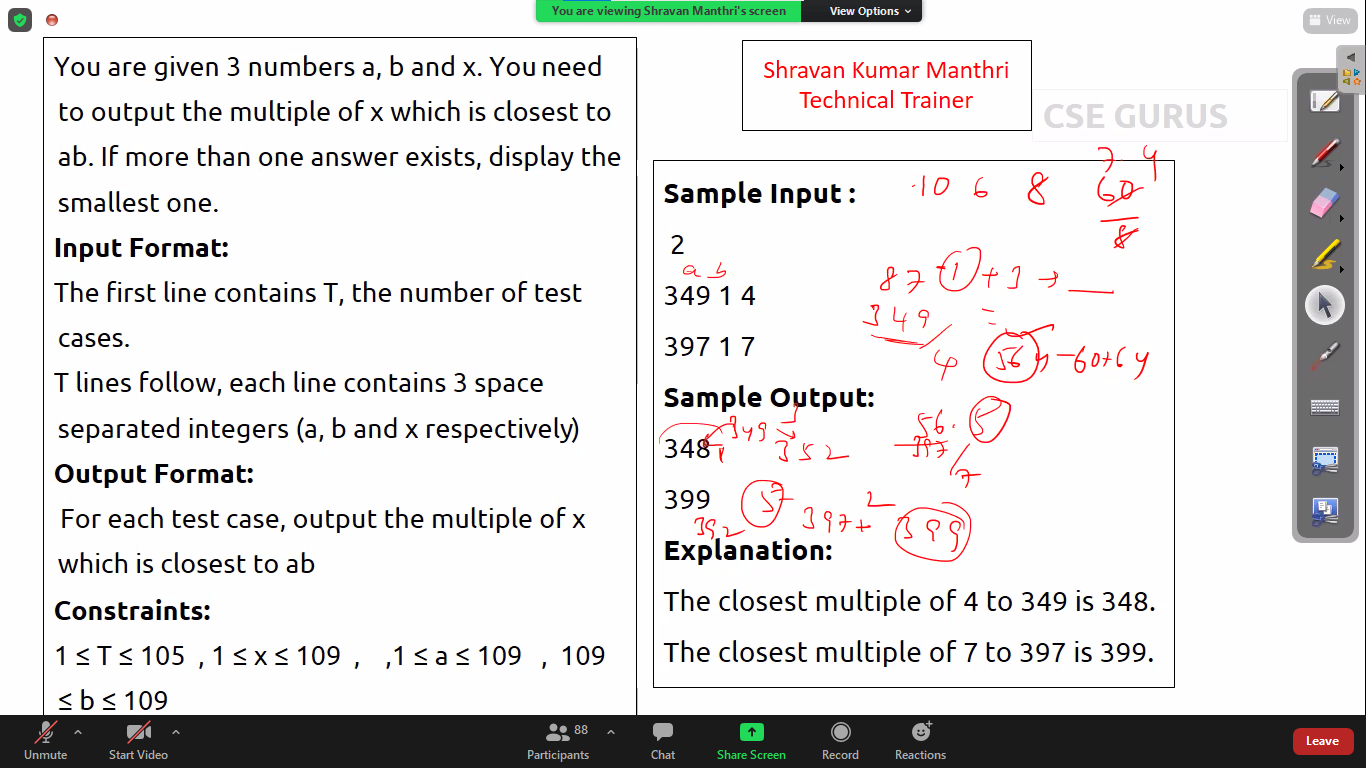
printf("Year:%d\n",a[i].year);

printf("Percentage:%f",a[i].percentage);

}

return 0;

}



#include<stdio.h>

int main(){

int t,a,b,x,mul,temp,flag;

scanf("%d",&t);

while(t--){

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

mul=a\*b;

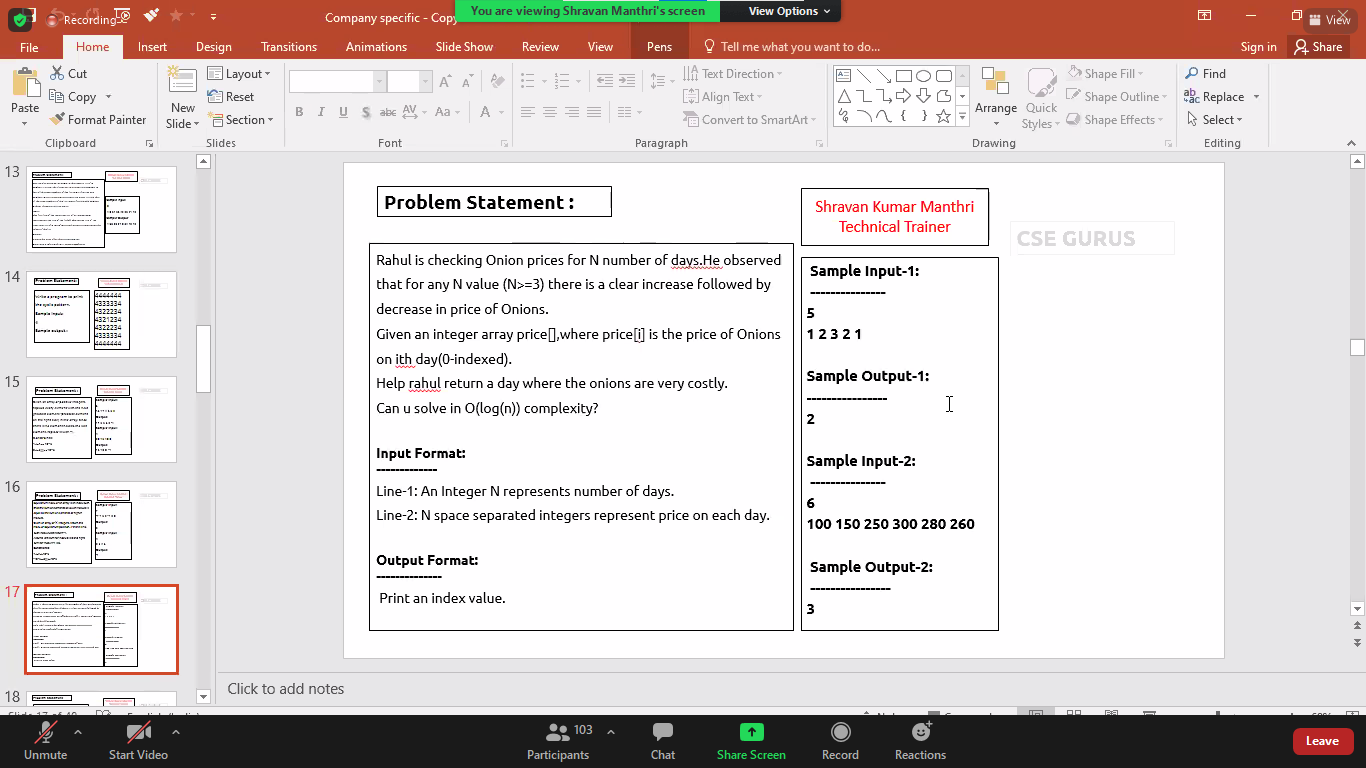
temp=mul/x;

flag=((temp+1)\*x-mul)>(mul-(x\*temp))?(x\*temp):(temp+1)\*x;

printf("%d",flag);

}

}



#include <stdio.h>

int main()

{

int n;

scanf("%d",&n);

int a[n],first=0,last=n-1,mid;

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

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

}

while(first<=last){

mid=(first+last)/2;

else if(a[mid]>a[mid+1] && a[mid]>a[mid-1]){

printf("%d",mid);

break;

}

else if(a[mid]>a[mid+1]){

last=mid-1;

}

else{

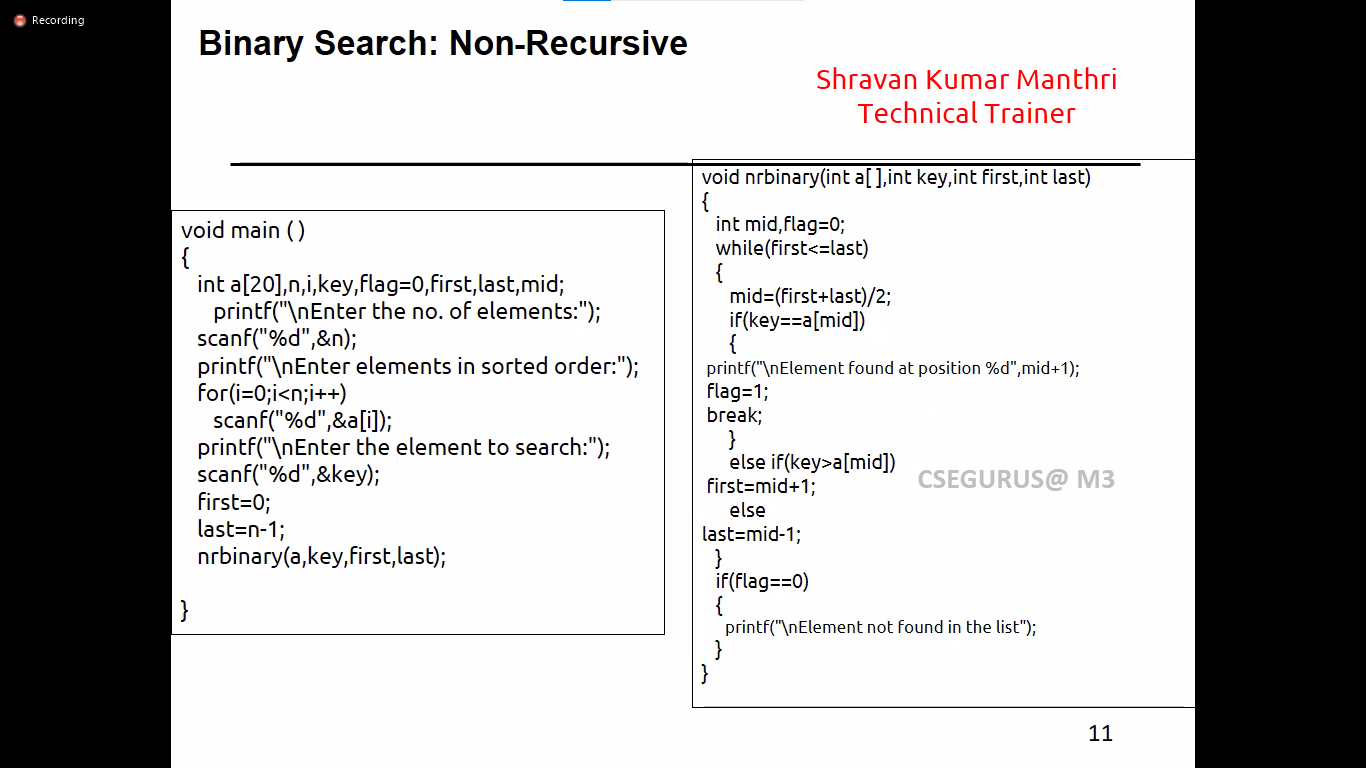
first=mid+1;

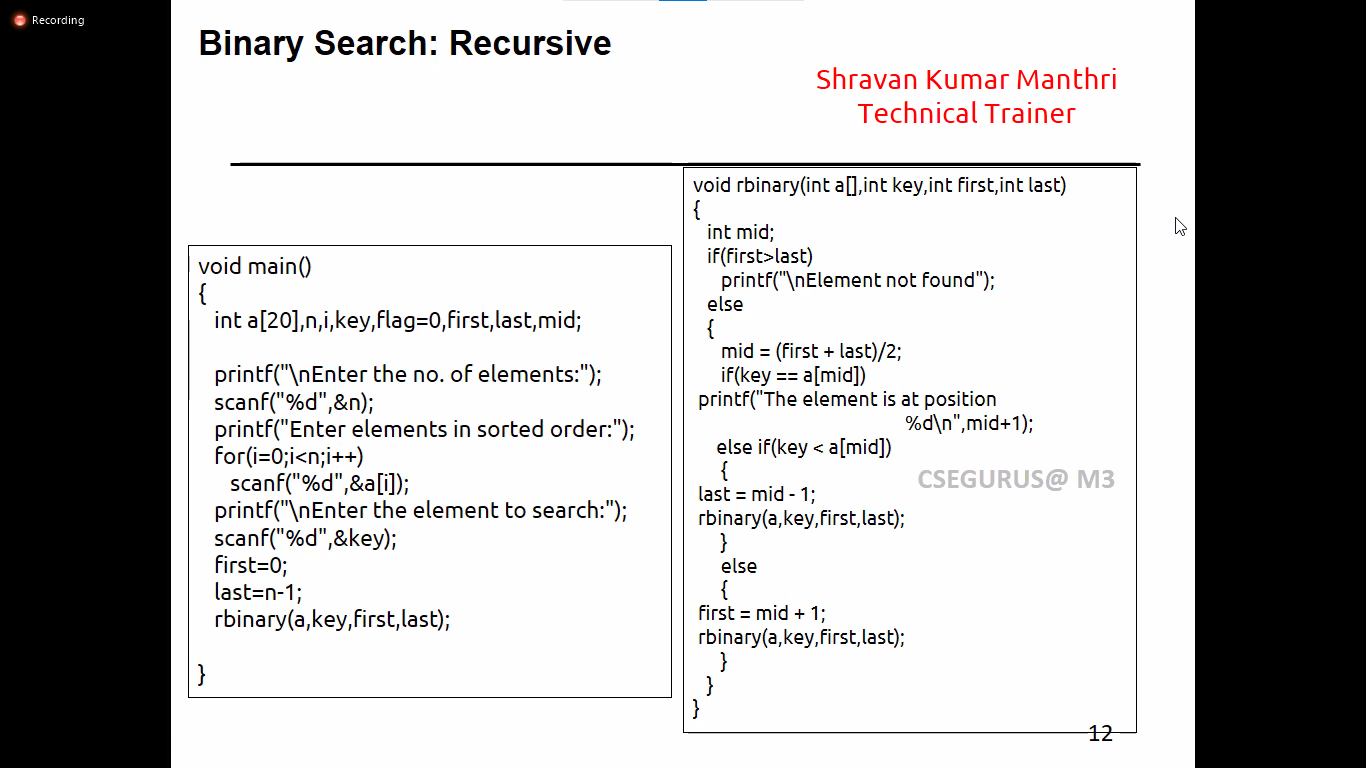
}

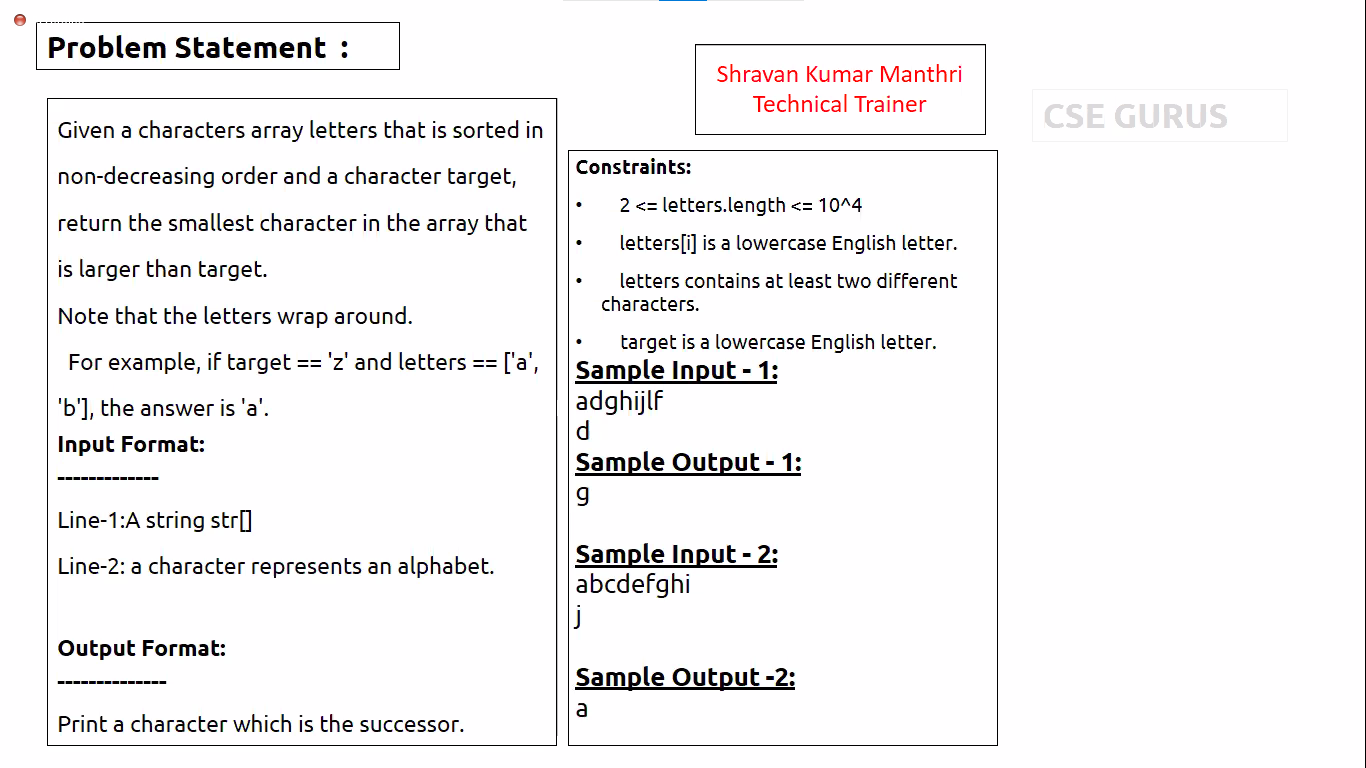
}

return 0;

}







char nextgreater(char [],char);

void main()

{

char str[2000],target;

gets(str);

scanf("%c",&target);

printf("%c",nextgreater(str,target));

}

char nextgreater(char str[],char t)

{

int l,h,n,i,mid;

for(n=0;str[n]!='\0';n++);

l=0;

h=n;

while(l<h)

{

mid=(l+h)/2;

if(str[mid]<=t)

l=mid+1;

else

h=mid;

}

return str[l%n];

}

(Or)

#include <stdio.h>

int main()

{

char str[100];

char target;

int n,a[26]={0};

scanf("%s ",str);

scanf("%c",&target);

for(n=0;str[n]!=0;n++){

a[str[n]-'a']=1;

}

for(int i=(target-'a')+1;;i=(i+1)%26){

if(a[i]==1){

printf("%c",i+97);

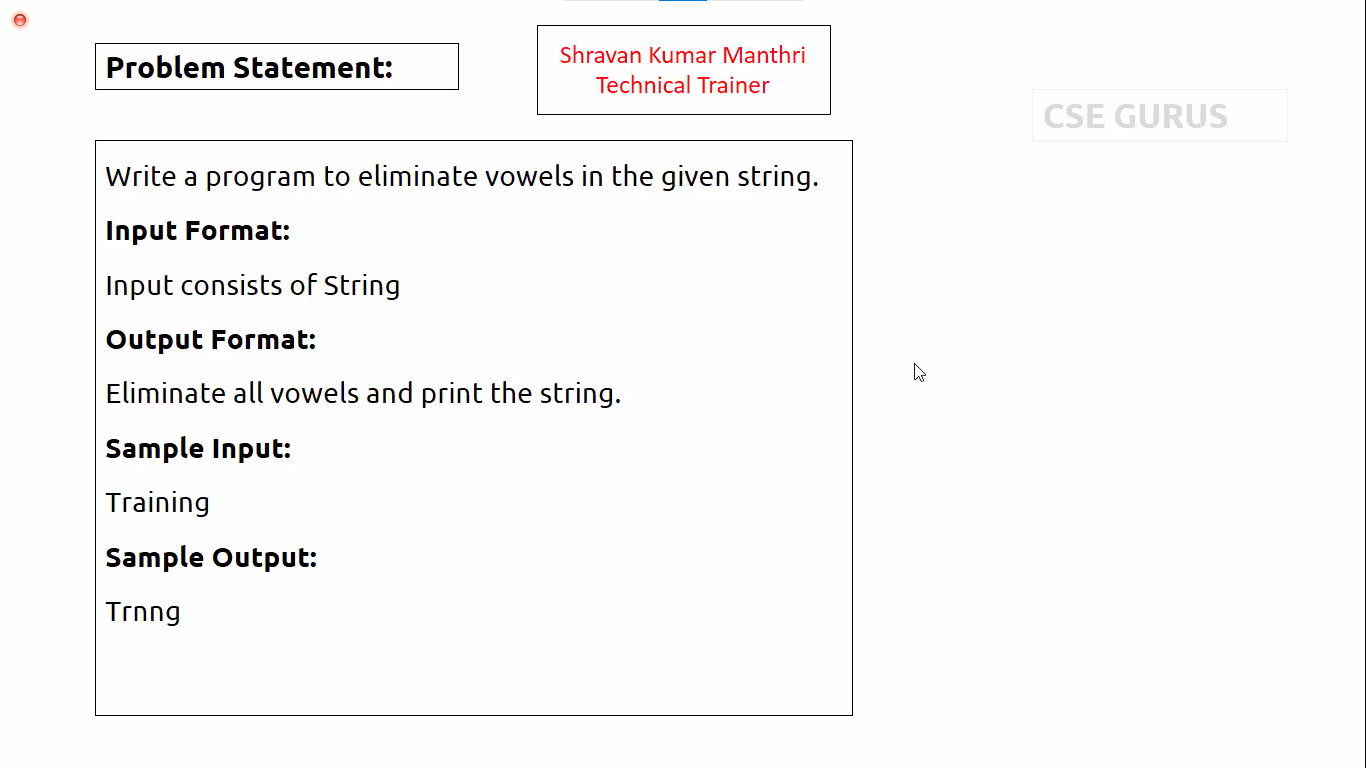
break;

}

}

return 0;

}



#include <stdio.h>

char\* remove\_vowels(char\* str){

int n,j=0;

for(n=0;str[n]!='\0';n++);

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

if(str[i]!='a' && str[i]!='e' && str[i]!='i' && str[i]!='o' && str[i]!='u'){

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

}

}

str[j]='\0';

return str;

}

int main()

{

char str[100];

scanf("%s",str);

printf("%s",remove\_vowels(str));

}