ASSIGNMENT NO 1

Q1.write a c program to input a element and search if it is present or not and print its frequency.

#include<stdio.h>

int main()

{

int a[10],n,i,s,f=0,fr=0;

printf("Enter the element");

scanf("%d",&n);

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

{

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

}

printf("Enter element to be searched");

scanf("%d",&s);

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

{

if(a[i]==s)

{

f=1;

fr++;

}

}

if(f==1)

{

printf("%d is present in the array %d times",s,fr);

}

else

{

printf( "%d is not present in array",s);

}

return 0;

}

Q2. Write a c program to delete duplicate element from an array.

#include <stdio.h>

int main()

{

int a[50],i,j,k,n;

printf("Enter the size");

scanf("%d",&n);

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

{

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

}

for(i=0;i<n;i++) //remove duplicate element

{

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

{

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

{

for(k=j;k<n-1;k++) //shifts element to fill the gap

{

a[k]=a[k+1];

}

n--; //reduces the size of array

j--; //ajust the inner loop index

}

}

}

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

{

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

}

return 0;

}

Q3.Write a c program to search largest and smallest value in 1D array.

#include <stdio.h>

int main()

{

int a[50],i,n,large,small;

printf("Enter the size");

scanf("%d",&n);

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

{

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

}

large=a[0];

small=a[0];

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

{

if(a[i]>large)

{

large=a[i];

}

if(a[i]<small)

{

small=a[i];

}

}

printf("The largest element is %d",large);

printf("\nThe smallest element is %d",small);

return 0;

}

Q4.Write a c program to multiply the matrix.

#include <stdio.h>

int main()

{

int a[5][5],b[5][5],c[5][5],i,j,k,m1,n1,m2,n2;

printf("Input rows and columns");

scanf("%d%d%d%d",&m1,&n1,&m2,&n2);

if(n1!=m2)

{

printf("Multiplication not possible");

}

else

{

for(i=0;i<m1;i++)

{

for(j=0;j<n1;j++)

{

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

}

}

for(i=0;i<m1;i++)

{

for(j=0;j<n1;j++)

{

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

}

}

for(i=0;i<m1;i++)

{

for(j=0;j<n2;j++)

{

for(k=0;k<n1;k++)

{

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

}

}

}

}

return 0;

}

Q5.Write a c program to transpose a matrix.

#include <stdio.h>

int main()

{

int a[5][5],i,j,m1,n1,temp;

printf("Input rows and columns");

scanf("%d%d",&m1,&n1);

for(i=0;i<m1;i++)

{

for(j=0;j<n1;j++)

{

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

}

}

for(i=0;i<m1;i++)

{

for(j=0;j<n1;j++)

{

temp=a[i][j];

a[i][j]=a[j][i];

a[j][i]=temp;

}

}

for(i=0;i<n1;i++)

{

for(j=0;j<m1;j++)

{

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

}

printf("\n");

}

return 0;

}

Q6.Write a code to print the sum of elements present in principal diagonal separately.

#include<stdio.h>

int main()

{

int a[5][5],i,j,m,n,sump=0,sum=0;

printf("Input rows and column");

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

if(m!=n)

{

printf("Not a square matrix");

}

else

{

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

{

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

{

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

}

}

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

{

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

{

if(i==j)

{

sump=sump+a[i][j];

}

if((i+j)==(m-1))

{

sum=sum+a[i][j];

}

}

}

}

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

return 0;

}

Q7.Write a c program to print smallest number of principal diagonal and largest of secondary diagonal.

#include<stdio.h>

int main()

{

int a[5][5],i,j,m,n,max,min;

printf("Input rows and column");

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

if(m!=n)

{

printf("Not a square matrix");

}

else

{

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

{

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

{

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

}

}

max=a[0][m-1];

min=a[0][0];

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

{

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

{

if(i==j)

{

if(min>a[i][j])

{

min=a[i][j];

}

}

if((i+j)==(m-1))

{

if(max<a[i][j])

{

max=a[i][j];

}

}

}

}

}

printf("%d",min);

printf("%d",max);

return 0;

}

Q8Write a code to print the sum of all the elements in principal and secondary diagonal.

#include<stdio.h>

int main()

{

int a[5][5],i,j,m,n,sump=0,sum=0,sums;

printf("Input rows and column");

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

if(m!=n)

{

printf("Not a square matrix");

}

else

{

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

{

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

{

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

}

}

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

{

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

{

if(i==j)

{

sump=sump+a[i][j];

}

if((i+j)==(m-1))

{

sum=sum+a[i][j];

}

}

}

}

if(m%2==0)

{

sums=sump+sum;

}

else

{

sums=sump+sum+a[m/2][n/2];

}

printf("%d",sums);

return 0;

}

Q9.Write a c program to print all even numbers present in upper triangular matrix.

#include<stdio.h>

int main()

{

int a[5][5],i,m,n,j;

printf("Enter rows and column");

scanf("%d%d",&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++)

{

if(i<=j&&a[i][j]%2==0)

{

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

}

else

{

printf("%d ", 0);

}

}

printf("\n");

}

return 0;

}

Q10. Write a c program to print all even numbers present in lower triangular matrix.

#include<stdio.h>

int main()

{

int a[5][5],i,m,n,j;

printf("Enter rows and column");

scanf("%d%d",&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++)

{

if(i>=j&&a[i][j]%2==0)

{

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

}

else

{

printf("%d ", 0);

}

}

printf("\n");

}

return 0;

}

Q11.Write a c program to find lcm of two numbers usings recursions.

#include <stdio.h>

int lcm(int a, int b);

int main()

{

int num1, num2, LCM;

printf("Enter any two numbers to find lcm: ");

scanf("%d%d", &num1, &num2);

if(num1 > num2)

LCM = lcm(num2, num1);

else

LCM = lcm(num1, num2);

printf("LCM of %d and %d = %d", num1, num2, LCM);

return 0;

}

Q12.Write a c program to print all strong numbers between given interval using functions.

#include <stdio.h>

long long fact(int num);

void StrongNumbers(int,int);

int main()

{

int start, end;

printf("Enter the lower limit to find strong number: ");

scanf("%d", &start);

printf("Enter the upper limit to find strong number: ");

scanf("%d", &end);

printf("All strong numbers between %d to %d are: \n", start, end);

StrongNumbers(start, end);

return 0;

}

void StrongNumbers(int start, int end)

{

long long sum;

int num;

while(start != end)

{

sum = 0;

num = start;

while(num != 0)

{

sum += fact(num % 10);

num /= 10;

}

if(start == sum)

{

printf("%d, ", start);

}

start++;

}

}

long long fact(int num)

{

if(num == 0)

return 1;

else

return (num \* fact(num-1));

}

Q13.Write a c program GCD(HCF) of a two numbers using recursion.

#include <stdio.h>

int GCD(int,int);

int main()

{

int x,y,res;

printf("Enter the values");

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

res=GCD(x,y);

printf("%d",res);

return 0;

}

int GCD(int div,int dd)

{

int rem;

rem=dd%div;

if(rem==0)

{

return div;

}

else

{

GCD(rem,div);

}

}

Q14.Write a c program to input a decimal number and convert it to binary using function.

#include<stdio.h>

long Binary(int);

int main()

{

long res;

int n;

printf(" Input any decimal number : ");

scanf("%d",&n);

res = Binary(n);

printf("\n The Binary value is : %ld\n\n",res);

return 0;

}

long Binary(int n)

{

long res=0,rem,f=1;

while(n != 0)

{

rem = n % 2;

res = res + rem \* f;

f = f \* 10;

n = n / 2;

}

return res;

}

Q15.Write a c program to solve tower of Hanoi using recursive method.

#include <stdio.h>

void TOH(int n, char A, char C, char B);

int main()

{

int n ;

printf("Enter number of disks: ");

scanf("%d", &n);

TOH(n, 'A','C','B');

return 0;

}

void TOH(int n, char A, char C, char B)

{

if (n == 1)

{

printf("\n Move disk 1 from rod %c to rod %c",A ,C );

return;

}

TOH(n-1, A, B, C);

printf("\n Move disk %d from rod %c to rod %c", n, A, C);

TOH(n-1, B, C,A);

}

Q16.Write a c program to input matrix and sort its column in ascending order using functions.

#include <stdio.h>

void s(int m, int n, int ar[m][n])

{

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

{

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

{

for (int k = i + 1; k < m; k++)

{

if (ar[i][j] > ar[k][j])

{

int temp = ar[i][j];

ar[i][j] = ar[k][j];

ar[k][j] = temp;

}

}

}

}

}

int main()

{

int m, n;

printf("Enter rows");

scanf("%d", &m);

printf("Enter columns ");

scanf("%d", &n);

int ar[m][n];

printf("elementt \n");

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

{

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

{

printf("elementt at %d, %d ", i+1, j+1);

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

}

}

s(m, n, ar);

printf("2D array\n");

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

{

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

{

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

}

printf("\n");

}

return 0;

}

Q17.Write a c program to input a matrix and sort its rows in ascending order using functions.

#include <stdio.h>

void s(int n, int row[])

{

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

{

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

{

if (row[j] > row[j + 1])

{

int temp = row[j];

row[j] = row[j + 1];

row[j + 1] = temp;

}

}

}

}

int main()

{

int m, n;

printf("rows");

scanf("%d", &m);

printf("col ");

scanf("%d", &n);

int ar[m][n];

printf("elements\n");

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

{

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

{

printf("elementt at %d, %d ", i+1, j+1);

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

}

}

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

{

s(n, ar[i]);

}

printf("2D array\n");

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

{

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

{

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

}

printf("\n");

}

return 0;

}

Q18.Write a c program to multiply the matrix using function.

#include<stdio.h>

void multiply(int a[5][5],int b[5][5],int ,int ,int );

void main()

{

int a[5][5],b[5][5];

int i,j,k,m,n,p;

printf("Enter the number of rows and columns for 1st matrix\n");

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

printf("Enter the elements of the 1st matrix\n");

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

{

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

{

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

}

}

//no of col of 1st mat = no of rows of 2nd mat

printf("Enter the number of columns for 2nd matrix\n");

scanf("%d",&p);

printf("Enter the elements of the 2nd matrix\n");

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

{

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

{

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

}

}

printf("The 1st matrix\n");

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

{

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

{

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

}

printf("\n");

}

printf("The 2nd matrix\n");

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

{

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

{

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

}

printf("\n");

}

multiply(a,b,m,n,p);

}

void multiply(int a[5][5],int b[5][5],int m,int n,int p)

{

int mul[5][5],i,j,k;

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

{

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

{

mul[i][j]=0;

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

{

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

}

}

}

printf("The resultant matrix formed on multiplying the two matrices\n");

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

{

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

{

printf("%d\t",mul[i][j]);

}

printf("\n");

}

}