

**MCA SEC-C**

**NAME :- RITIK KAUSHIK**

**ROLL NO :- 48**

**UNIVERSITY ROLL NO :- 2284200171**

**SUBJECT :- Data Structure Algorithm**

**SEMESTER :- First Semester**

**SUBJECT TEACHER : - Dr. Siddharth Sir**

**INDEX**

**1 : - Program of Array Traversal**

**2 :- Program of insert an element**

**In Array**

**3 :- Program of delete an element**

**From Array**

**4 :- Program of Sorting of Array**

**Elements**

**5 :- Program of Array Multiplication**

**6 :- Program of Array Addition**

**7 :- Program of Searching an**

**Element in Array**

**8 : - Program Of Queue**

Q1- WAP to traverse the elements of matrix Array

#include<stdio.h>

int main()

{

int a[3][3],i,j;

printf("Enter 9 elements of matrix\n");

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

{

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

{

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

}

}

printf("Matrix is :\n");

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

{

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

{

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

}

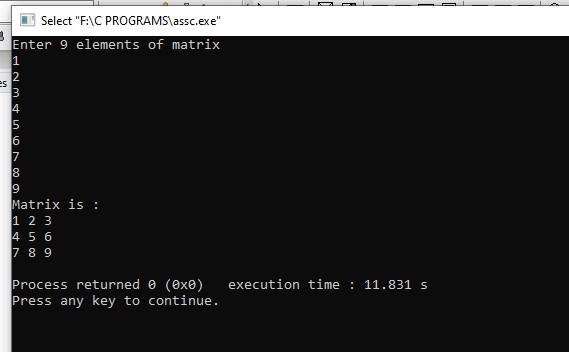
printf("\n");

}

return 0;

}

**Output**

****

Q2-WAP to insert an element in an array

#include<stdio.h>

int main()

{

int a[50],i,j,n,p,v;

printf("Enter Size of Array\n");

scanf("%d",&n);

printf("Enter %d elements\n",n);

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

{

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

}

printf("Entered Elements are\n");

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

{

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

}

printf("\n");

printf("Enter element you want to insert and position number\n");

scanf("%d%d",&v,&p);

for(i=n-1;i>=p-1;i--)

{

a[i+1] = a[i];

}

a[p-1] = v;

printf("Final Elements are\n");

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

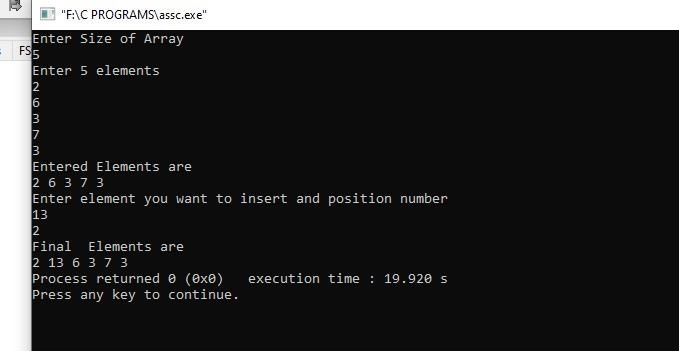
{

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

}

return 0;

**OUTPUT**

****

Q3- WAP to delete an element from array

#include<stdio.h>

int main()

{

int a[50],i,j,temp,n,p;

printf("Enter Size of Array\n");

scanf("%d",&n);

printf("Enter %d Elements\n",n);

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

{

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

}

printf(" Your Entered Array is\n");

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

{

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

}

printf("\n");

printf("Enter index number you want to delete\n");

scanf("%d",&p);

if(p>=n+1)

{

printf("WRONG POSITION ENTERED\n");

}

else

{

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

{

a[i]=a[i+1];

}

printf("Final Elements are\n");

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

{

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

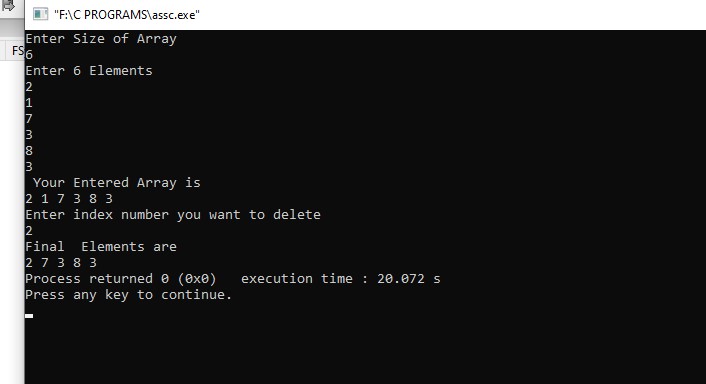
}

}

return 0;

}

**OUTPUT**

****

Q4- WAP to sort the array

#include<stdio.h>

int main()

{

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

printf("Enter Size of Array\n");

scanf("%d",&n);

printf("Enter %d Elements\n",n);

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

{

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

}

printf(" Your Entered Array is\n");

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

{

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

}

printf("\n");

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

{

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

{

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

{

temp = a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

printf(" Sorted Entered Array is\n");

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

{

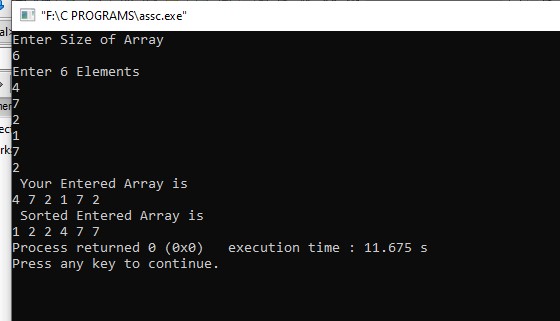
printf("%d ",a[i]);

}

return 0;

}

**OUTPUT**

****

Q5 – WAP to Multiply Two Matrices Using Array

#include<stdio.h>

int main()

{

int a[3][3],b[3][3],c[3][3],sum=0,k,i,j;

printf("Enter 9 elements of first matrix\n");

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

{

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

{

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

}

}

printf("Enter 9 elements of second matrix\n");

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

{

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

{

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

}

}

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

{

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

{

sum=0;

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

{

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

c[i][j]=sum;

}

}

}

printf("First matrix is\n");

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

{

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

{

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

}

printf("\n");

}

printf("Second matrix is\n");

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

{

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

{

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

}

printf("\n");

}

printf(" Matrix Multiplication is\n");

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

{

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

{

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

}

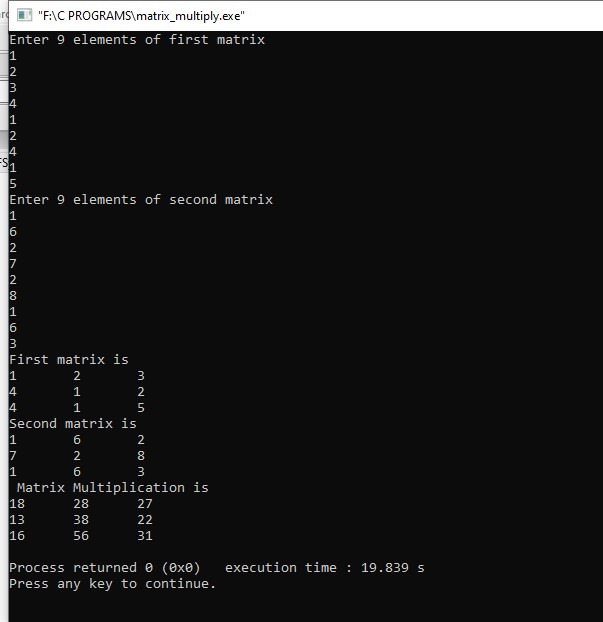
printf("\n");

}

return 0;

}

**OUTPUT**

****

Q6-WAP of Addition Of Two Matrices

#include<stdio.h>

int main()

{

int a[3][3],b[3][3],i,j;

printf("Enter 9 element of 1st matrix\n");

for(i=0;i<=2;i++)

{

for(j=0;j<=2;j++)

{

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

}

}

printf("\n");

printf("Enter 9 elements of 2nd matrix\n");

for(i=0;i<=2;i++)

{

for(j=0;j<=2;j++)

{

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

}

}

printf("Sum of two matrix are\n");

for(i=0;i<=2;i++)

{

for(j=0;j<=2;j++)

{

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

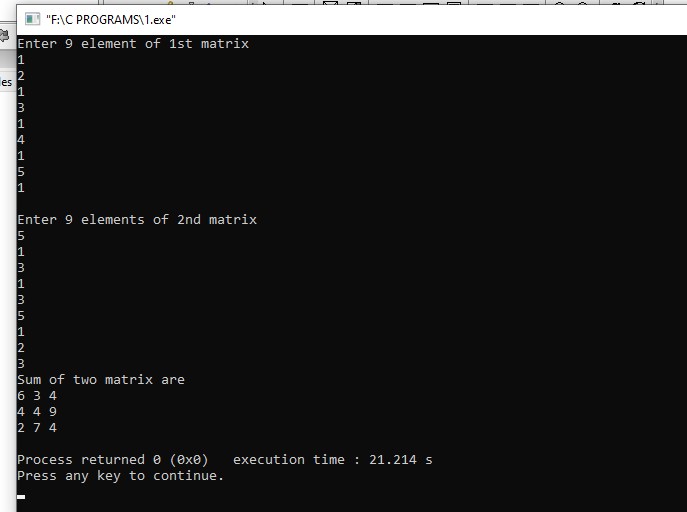
}

printf("\n");

}

}

OUTPUT



Q7- WAP to find an Search from Array

#include<stdio.h>

int main()

{

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

printf("Enter Size Of Array\n");

scanf("%d",&n);

printf("Enter %d Elements\n",n);

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

{

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

}

printf("Entered Elements are:\n");

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

{

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

}

printf("\n");

printf("Enter element You want to search\n");

scanf("%d",&e);

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

{

if(a[i]==e)

{

printf("Your element is %d position",i+1);

break;

}

}

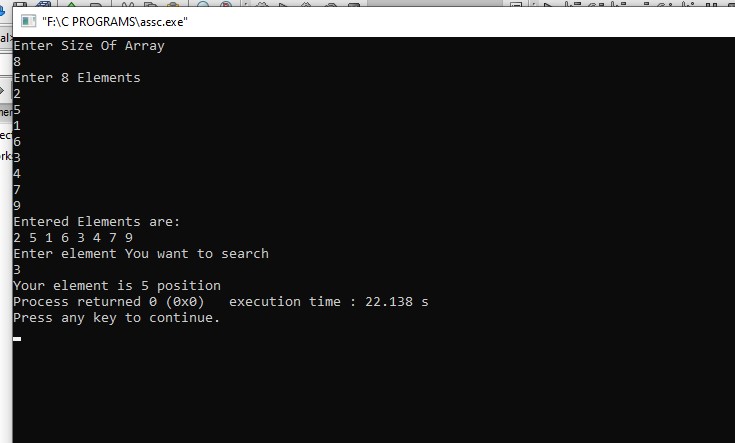
if(i==n)

printf(" Element not present\n");

return 0;

}

**OUTPUT**

****

Q8 WAP to implement Queue and its Operations

#include<iostream>

using namespace std;

class Queue

{

public:

int q[100];

int Front;

int rear;

int sizes;

Queue()

{

Front=-1;

rear=-1;

sizes=100;

}

void enque(int v)

{

if(rear==sizes-1)

{

cout<<"OverFlow\n";

}

else if(Front==-1 && rear==-1)

{

Front++;

rear++;

q[rear]=v;

}

else

{

rear++;

q[rear]=v;

}

}

void deeque()

{

if(Front==-1)

{

cout<<"Underflow\n";

}

else if(Front==rear)

{

cout<<q[Front]<<" deleted "<<endl;

Front=-1;

rear=-1;

}

else

{

cout<<q[Front]<<" deleted "<<endl;

Front++;

}

}

void print()

{

int i;

if(Front==-1)

{

cout<<"Queue is Empty\n";

}

else

{

for(i=Front;i<=rear;i++)

{

cout<<q[i]<<" "<<endl;

}

}

}

};

int main()

{

Queue q1;

cout<<"Insert Elements on Queue\n";

q1.enque(5);

q1.print();

cout<<endl;

q1.enque(6);

q1.print();

cout<<endl;

q1.enque(7);

q1.print();

cout<<endl;

q1.enque(8);

q1.print();

cout<<endl;

cout<<"Delete Elements From Queue\n";

q1.deeque();

q1.print();

cout<<endl;

q1.deeque();

q1.print();

cout<<endl;

q1.deeque();

q1.print();

cout<<endl;

q1.deeque();

q1.print();

cout<<endl;

q1.deeque();

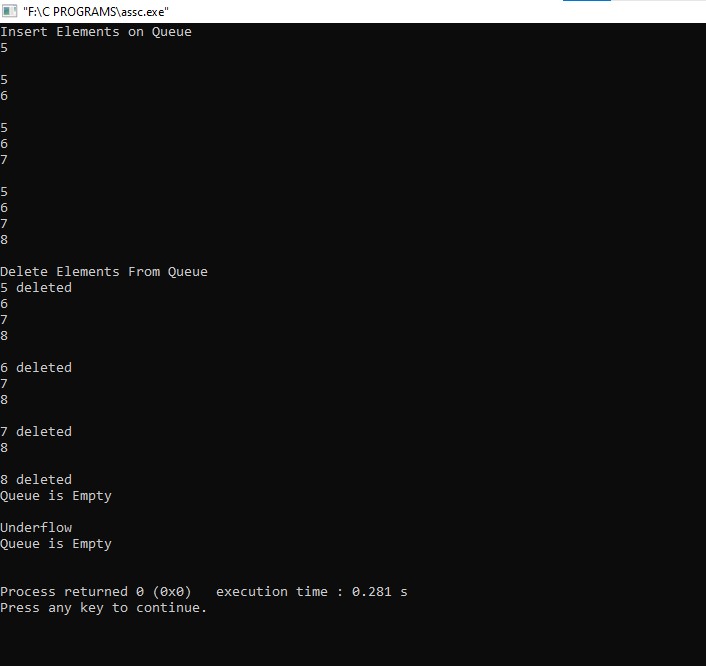
q1.print();

cout<<endl;

return 0;

}

**OUTPUT**

****