•

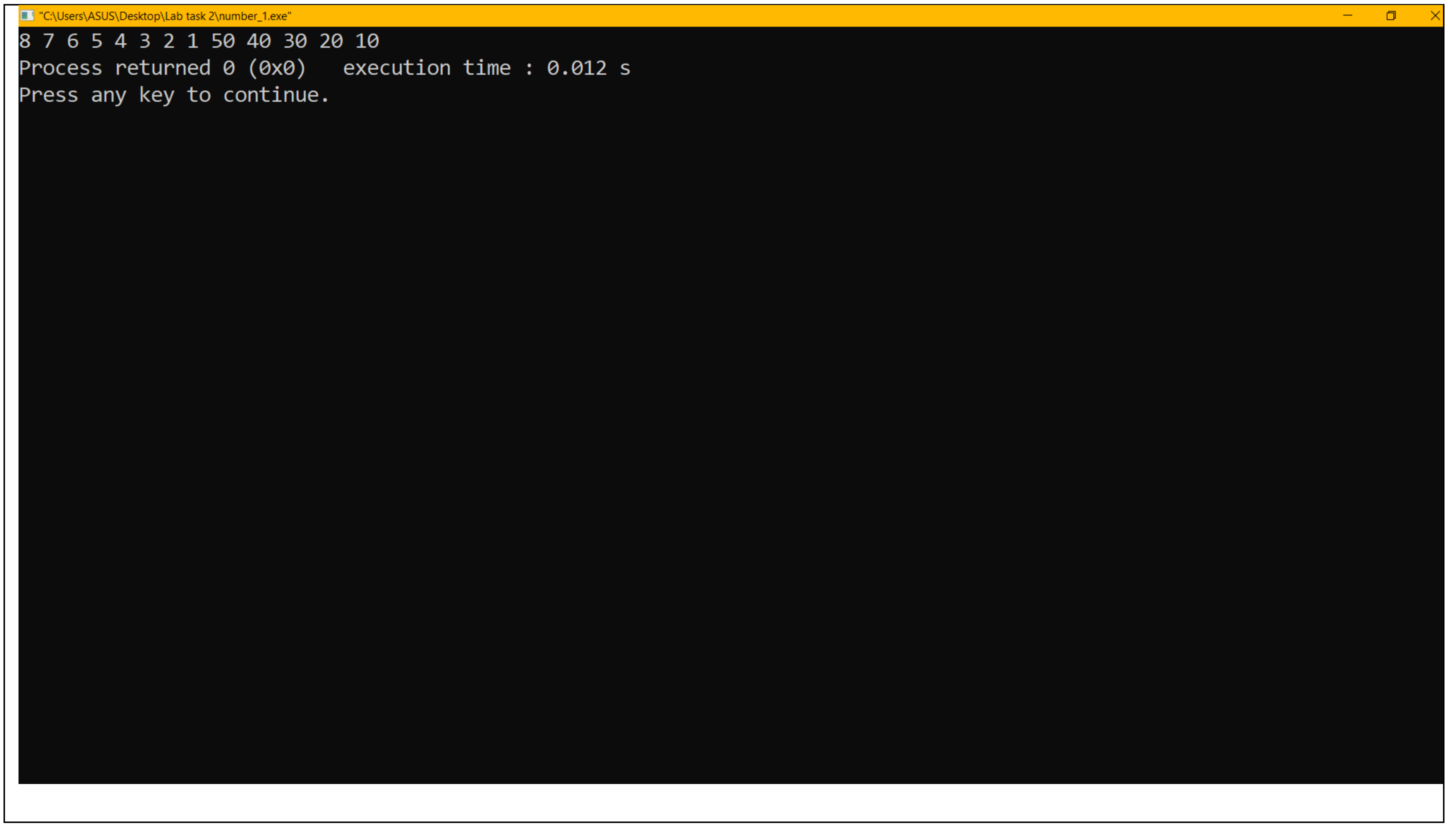
1. Initialize TWO integer arrays of different sizes. Merge the input arrays and create a new array. Then print the new array in reverse order.

For example,
Array_1 = **{10,20,30,40,50}**
Array_2 = **{1,2,3,4,5,6,7,8}**
Output: **8 7 6 5 4 3 2 1 50 40 30 20 10**

Your code here:

```
#include<iostream>
using namespace std;
int main()
{
    int arr1[5]={10,20,30,40,50};
    int size1=5;
   int arr2[8]={1,2,3,4,5,6,7,8};
  int size2=8;
  int k;
   int Msize=size1+size2;
  int marge[Msize];

for (int i=0;i<size1;i++)
    {
        marge[i]=arr1[i];
    }


  for (int i=0;i<size2;i++)
    {
        marge[i+5]=arr2[i];

    }


for(int i=Msize-1;i>=0;i--)
{
    cout<<marge[i]<<" ";
}

}
```

Your whole Screenshot here: (Console Output):



```
8 7 6 5 4 3 2 1 50 40 30 20 10
Process returned 0 (0x0)   execution time : 0.012 s
Press any key to continue.
```

2. Initialize TWO integer arrays **A** and **B** of different sizes. Make a new array with the common elements between **A** and **B**. Print the new array element(s). If there is no common element, output “No common element!”.

For example,

Scenario 1:

Array_1 = **{1,4,6,3,6,9}**

Array_2 = **{5,3,7,1,2,6}**

Output: **1 6 3**

Scenario 2:

Array_1 = **{1,4,6,3,6,9}**

Array_2 = **{5,8,7,12,21,63}**

Output: **No common element!**

Your code here:

Scenario _1

```
#include<iostream>
using namespace std;
int main()
{
    int arr1[6]={1,4,6,3,6,9};
    int size=6;
   int arr2[6]={5,3,7,1,2,6};

  int k,i,x;
  k=0;
  x=0;
   int Msize=12;
```

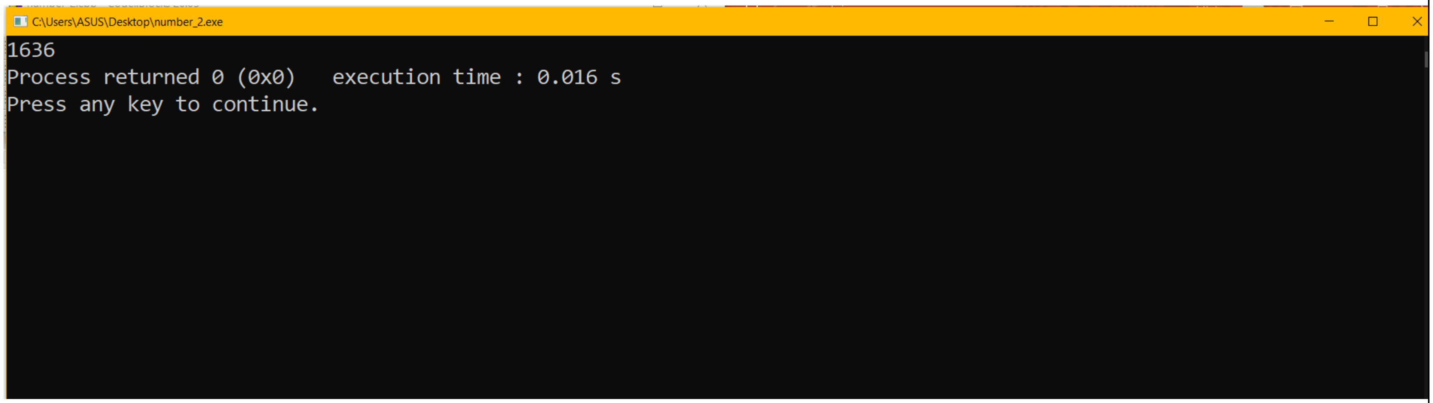
```
 int arr3[Msize];

  for (int i=0;i<size;i++)
  {
   int chk=arr1[i];
   for(int j=0; j<6;j++)
   {

     if(arr2[j]==chk)
     {
       arr3[x++]=chk;
     }
   }
  }
if (x!=0)
{
  for(i=0;i<x;i++)
{
  cout<<arr3[i];
}

}


else
  cout<<"No common element";

}
```

Scenario _2

```
#include<iostream>
using namespace std;
int main()
{
  int arr1[6]={1,4,6,3,6,9};
  int size=6;
 int arr2[6]={5,8,7,12,21,63};

 int k,i,x;
 k=0;
 x=0;
  int Msize=12;
 int arr3[Msize];

  for (int i=0;i<size;i++)
  {
   int chk=arr1[i];
   for(int j=0; j<6;j++)
   {

     if(arr2[j]==chk)
     {
       arr3[x++]=chk;
     }
   }
  }
if (x!=0)
{
  for(i=0;i<x;i++)
{
  cout<<arr3[i];
}

}


else
  cout<<"No common element";

}
```

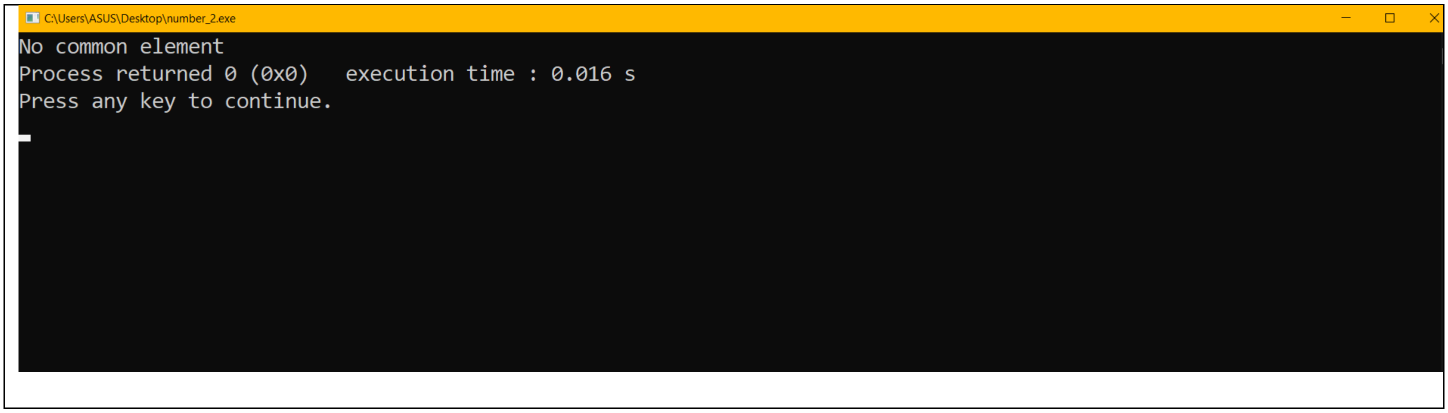
Your whole Screenshot here: (Console Output):

Scenario _1



```
1636
Process returned 0 (0x0)   execution time : 0.016 s
Press any key to continue.
```

Scenario _2



```
No common element
Process returned 0 (0x0)   execution time : 0.016 s
Press any key to continue.
```

3. Initialize an array. Size should be more than FIVE. Write you program to change the array in such a way so that there cannot be any duplicate element in the array anymore. Print the changed array. If the initialized array already had no duplicate elements from the beginning, output a message saying "Array already unique!";

For example,

Scenario 1:

Array_1 = **{1,4,6,3,6,9,1}**

Output: **1 4 6 3 9**

Scenario 2:

Array_1 = **{1,4,5,3,6,9}**

Output: **Array already unique!**

Your code here:

Scenario 1:

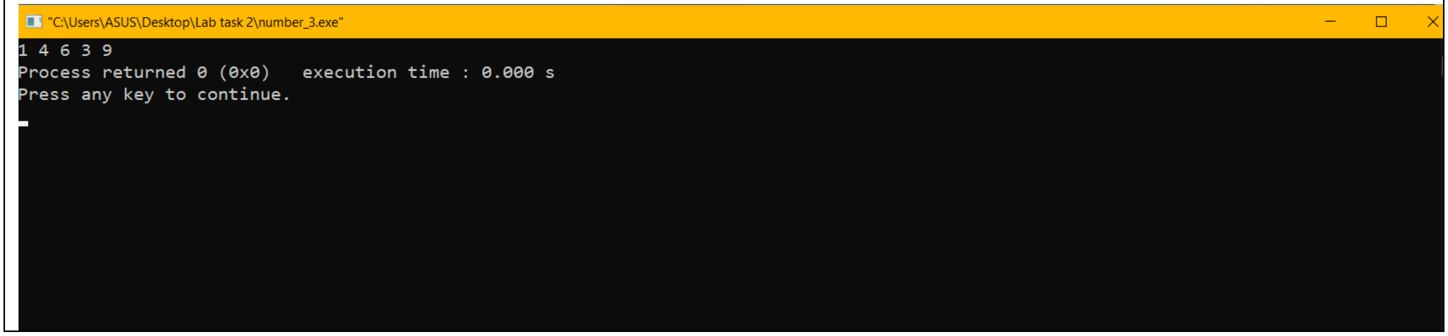
```
#include <iostream>
using namespace std;
int main() {
  int Array_1[12] = {1,4,6,3,6,9,1};
  int repeatEliment=0;
  for (int i=0;i<7;i++)
  {
    for(int j=i+1;j<7;j++)
    {
      if(Array_1[i]==Array_1[j])
      {
        for(int k=j;k<7;k++)
        {
          Array_1[k]=Array_1[k+1];
        }
        repeatEliment= repeatEliment+1;
      }
    }
  }
  if(repeatEliment==0)
  {
    cout<<"Array already unique!";
  }
  else{
   for(int t=0;t<5;t++)
   {
      cout<<Array_1[t]<<" ";
   }
}
}
```

Scenario 2:

```
#include <iostream>
using namespace std;
int main() {
   int Array_1[12] = {1,4,5,3,6,9};
   int repeatEliment=0;
   for (int i=0;i<7;i++)
   {
      for(int j=i+1;j<7;j++)
      {
         if(Array_1[i]==Array_1[j])
         {
            for(int k=j;k<7;k++)
            {
               Array_1[k]=Array_1[k+1];
            }
            repeatEliment= repeatEliment+1;
         }
      }
   }
   if(repeatEliment==0)
   {
      cout<<"Array already unique!";
   }
   else{
   for(int t=0;t<5;t++)
   {
      cout<<Array_1[t]<<" ";
   }
}
}
```

Your whole Screenshot here: (Console Output):

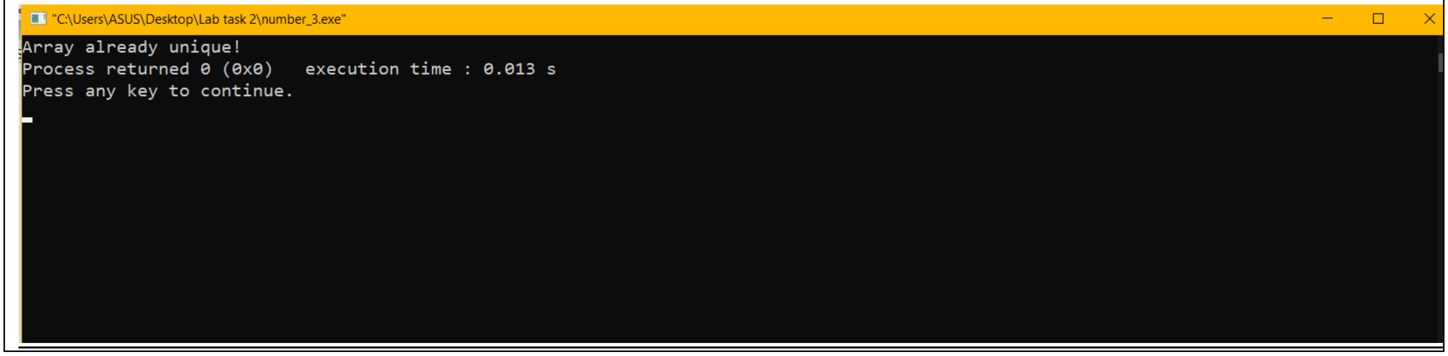
Scenario 1:



"C:\Users\ASUS\Desktop\Lab task 2\number_3.exe"

```
1 4 6 3 9
Process returned 0 (0x0)   execution time : 0.000 s
Press any key to continue.
```

Scenario 2:



```
Array already unique!
Process returned 0 (0x0)   execution time : 0.013 s
Press any key to continue.
```

4. Initialize an integer array **A** of size 10. Take an integer as input and print how many times that integer occurs in **A**.

For example,

Array_1 = **{8,4,6,1,6,9,6,1,9,8}**

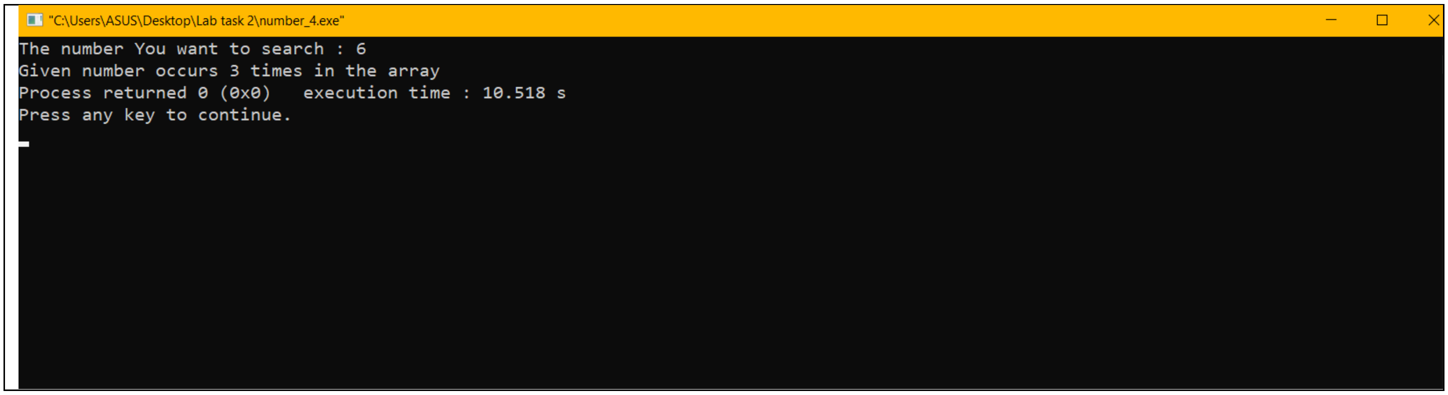
Output:

Input a number to search: 6
The number occurs 3 times in the array

Your code here:

```
#include<iostream>
using namespace std;
int main()
{
   int arr[10]={8,4,6,1,6,9,6,1,9,8};
   int x=10;
   int count=0;
   int n;
   cout<<"The number You want to search : ";
   cin>>n;
   for (int i=0;i<x;i++)
   {
      if(n==arr[i])
      {
         count=count+1;
      }
   }


   cout<<"Given number occurs "<<count<<" times in the array";
}
```

Your whole Screenshot here: (Console Output):



```
The number You want to search : 6
Given number occurs 3 times in the array
Process returned 0 (0x0)   execution time : 10.518 s
Press any key to continue.
```

5. Initialize an integer array of size 10. Print the number of time each element occurs in the array.

For example,

Array_1 = **{8,4,6,1,6,9,6,1,9,8}**

Output:

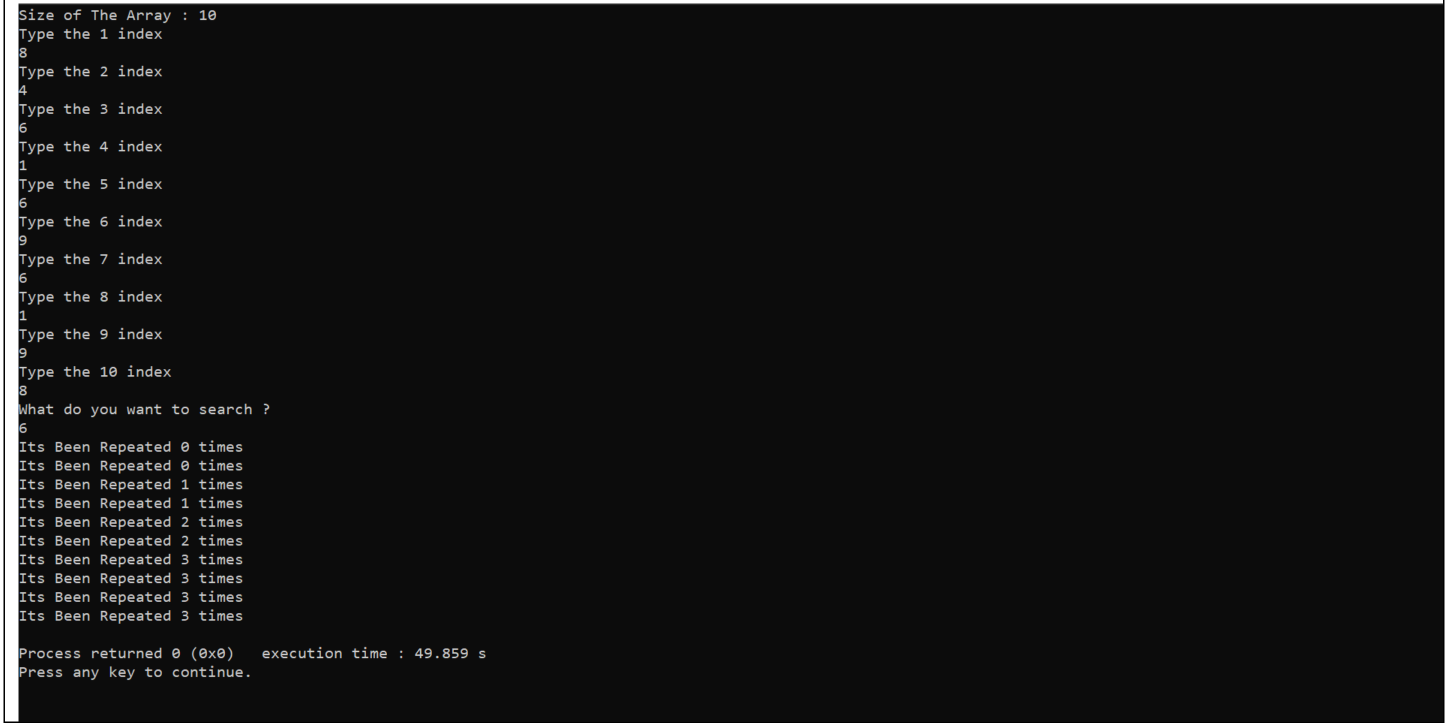
8 occurs = 2 times
4 occurs = 1 time
6 occurs = 3 times
1 occurs = 2 times
9 occurs = 2 times

Your code here:

```
#include <iostream>
using namespace std;
int main()
{
   int n;
   int count;
   int x;
   cout << "Size of The Array : ";
   cin >> n;
   int *arr = new int(n);
   count = 0;
   for (int i = 0; i < n; i++)
   {
      cout << "Type the " << i + 1 << " index" << endl;
      cin >> arr[i];
   }
   cout << "What do you want to search ?" << endl;
   cin >> x;
   for (int i = 0; i < n; i++)
   {
      if (arr[i] == x)
      {
         count++;
      }
      cout << "Its Been Repeated " << count << " times" << endl;
}
```

```
    return 0;
}
```

Your whole Screenshot here: (Console Output):



```
Size of The Array : 10
Type the 1 index
8
Type the 2 index
4
Type the 3 index
6
Type the 4 index
1
Type the 5 index
6
Type the 6 index
9
Type the 7 index
6
Type the 8 index
1
Type the 9 index
9
Type the 10 index
8
What do you want to search ?
6
Its Been Repeated 0 times
Its Been Repeated 0 times
Its Been Repeated 1 times
Its Been Repeated 1 times
Its Been Repeated 2 times
Its Been Repeated 2 times
Its Been Repeated 3 times
Its Been Repeated 3 times
Its Been Repeated 3 times
Its Been Repeated 3 times

Process returned 0 (0x0)   execution time : 49.859 s
Press any key to continue.
```

6. Initialize a matrix of minimum 3x4 (row x column) size. Output its transpose matrix.

For example,

Matrix_1:

1 6 7 9

2 4 8 5

3 1 9 4

Output:

1 2 3

6 4 1

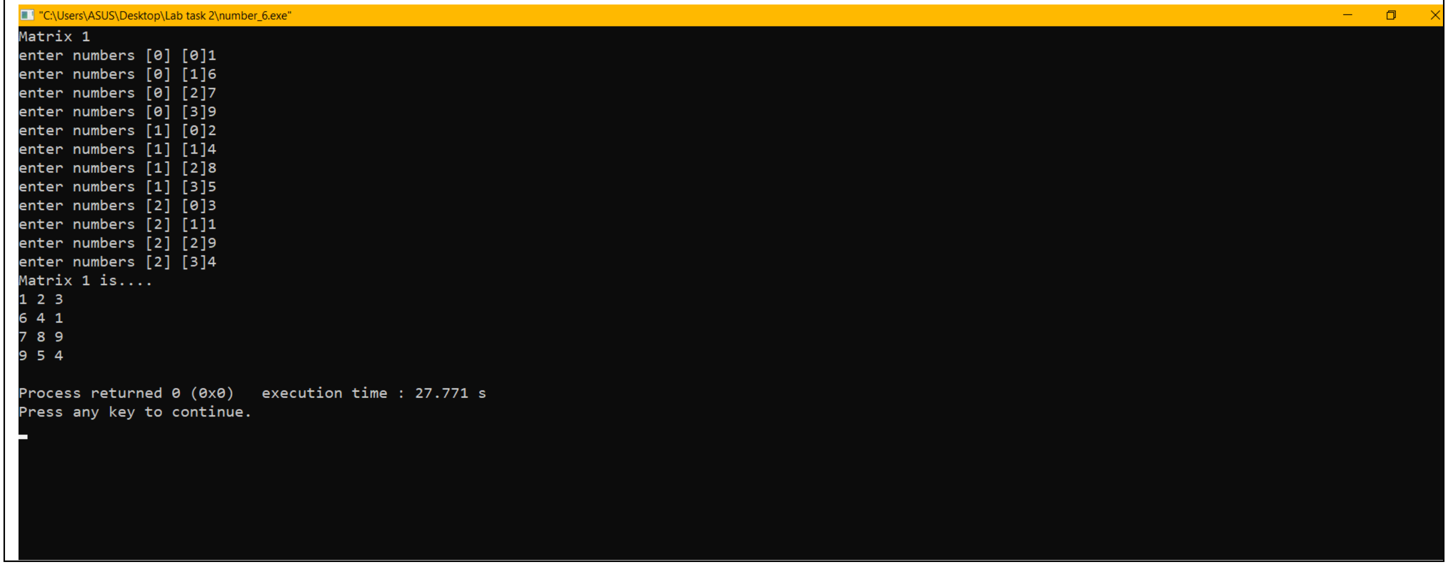
7 8 9

9 5 4

Your code here:

```
#include<iostream>
using namespace std;

int main()
{
    int a[3] [4],transpose[3] [4],i,j;
    cout<<"Matrix 1"<<endl;

    for(i=0;i<3;i++)
    {
        for(j=0;j<4;j++)
        {
            cout<<"enter numbers ["<<i<<"] ["<<j<<"]";
            cin>>a[i][j];
        }
    }

    cout<<"Matrix 1 is...."<<endl;
    for(i=0;i<4;i++)
    {
        for(j=0;j<3;j++)
        {
            cout<<a[j][i]<<" ";
        }
        cout<<endl;
    }
}
```

Your whole Screenshot here: (Console Output):



```
Matrix 1
enter numbers [0] [0]1
enter numbers [0] [1]6
enter numbers [0] [2]7
enter numbers [0] [3]9
enter numbers [1] [0]2
enter numbers [1] [1]4
enter numbers [1] [2]8
enter numbers [1] [3]5
enter numbers [2] [0]3
enter numbers [2] [1]1
enter numbers [2] [2]9
enter numbers [2] [3]4
Matrix 1 is....
1 2 3
6 4 1
7 8 9
9 5 4

Process returned 0 (0x0)   execution time : 27.771 s
Press any key to continue.
```