

# MADHAV INSTITUTE OF TECHNOLOGY AND SCIENCE



## DS lab ASSIGNMENTS :

Submitted to:

Prof. Lav Upadhyay

Submitted by:

Shalvi Singhal

CSE-2 nd year(III sem)

0901CS191112

## **INDEX:**

**Q.1 WAP for Fibonacci Series for fixed number and n numbers**

**Q.2 WAP for Armstrong of number**

**Q.3 WAP for Average of number.**

**Q.4 Develop following two program**

**a. GCD using simple school level algorithm**

**b. GCD using Euclidean Algorithm**

**Q5. (a) WAP to find largest number in an integer type array of size N where N and data elements are**

**taken as input from the user.**

**(b) WAP to find sum of all the numbers stored in a float type array of size N, where N and data**

**elements are taken as input from the user.**

**(c) WAP which takes N distinct integers as input in ascending order from the user and stores them**

**in a dynamically created array of size N. Than it performs search operation for target value X entered by user according to the search technique chosen by user. if user press 1 than it performs linear search operation and if user press 2 than it performs binary search operation.**

**Note:- result of the search operation should display index number of the array at which target**

**value X is present and number of comparisons it took to perform the search.**

**Q6. (a) WAP for reversing the element in array**

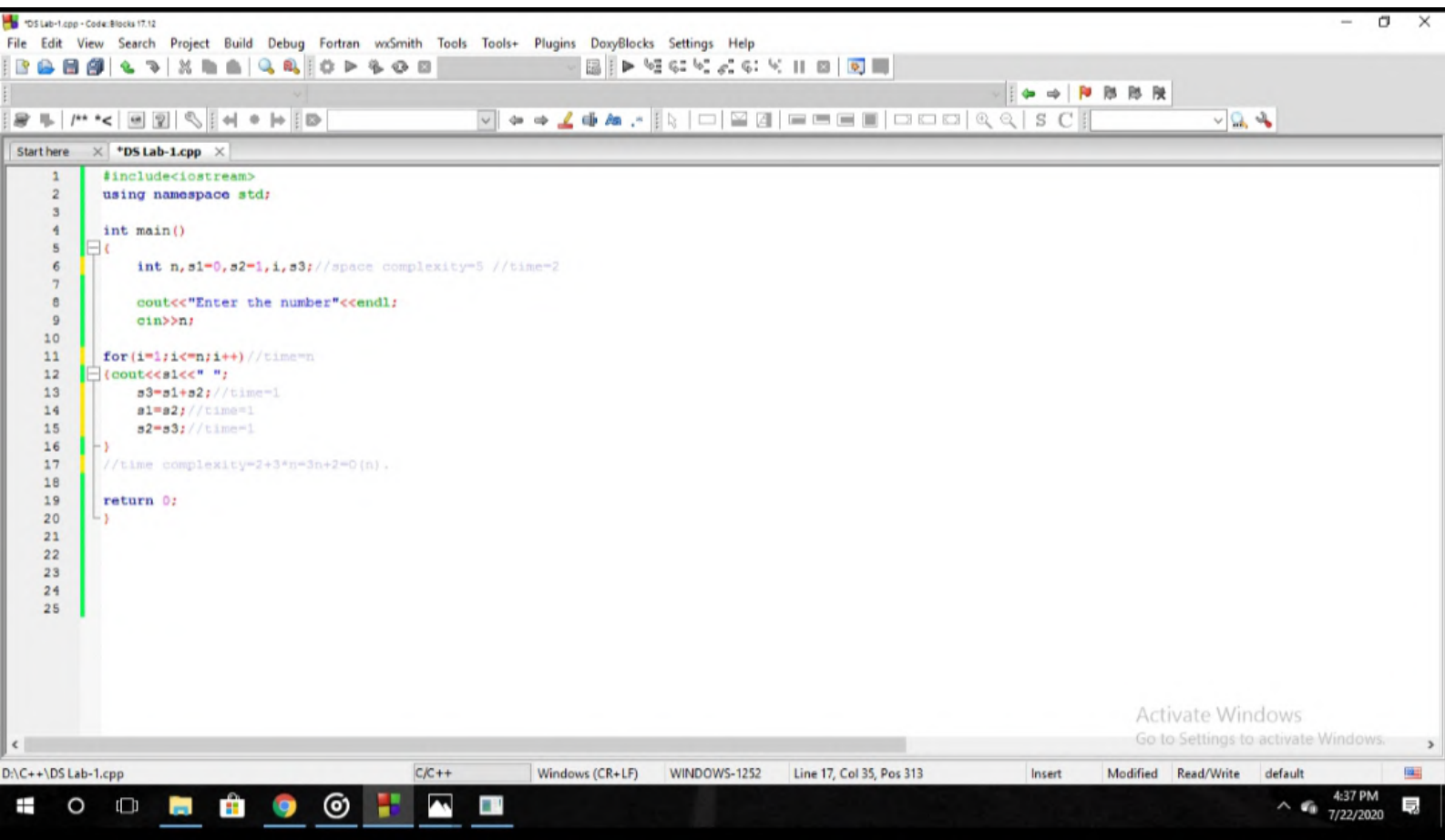
**(b) WAP for count odd and even numbers in array**

**Q7. (a) WAP to search an element in the Doubly Linked List**

**(b) WAP to search an element in the Circular Linked List.**

**Q.8 Implement Stack using Singly Linked List**

Q.1 WAP for Fibonacci Series for fixed number and n numbers



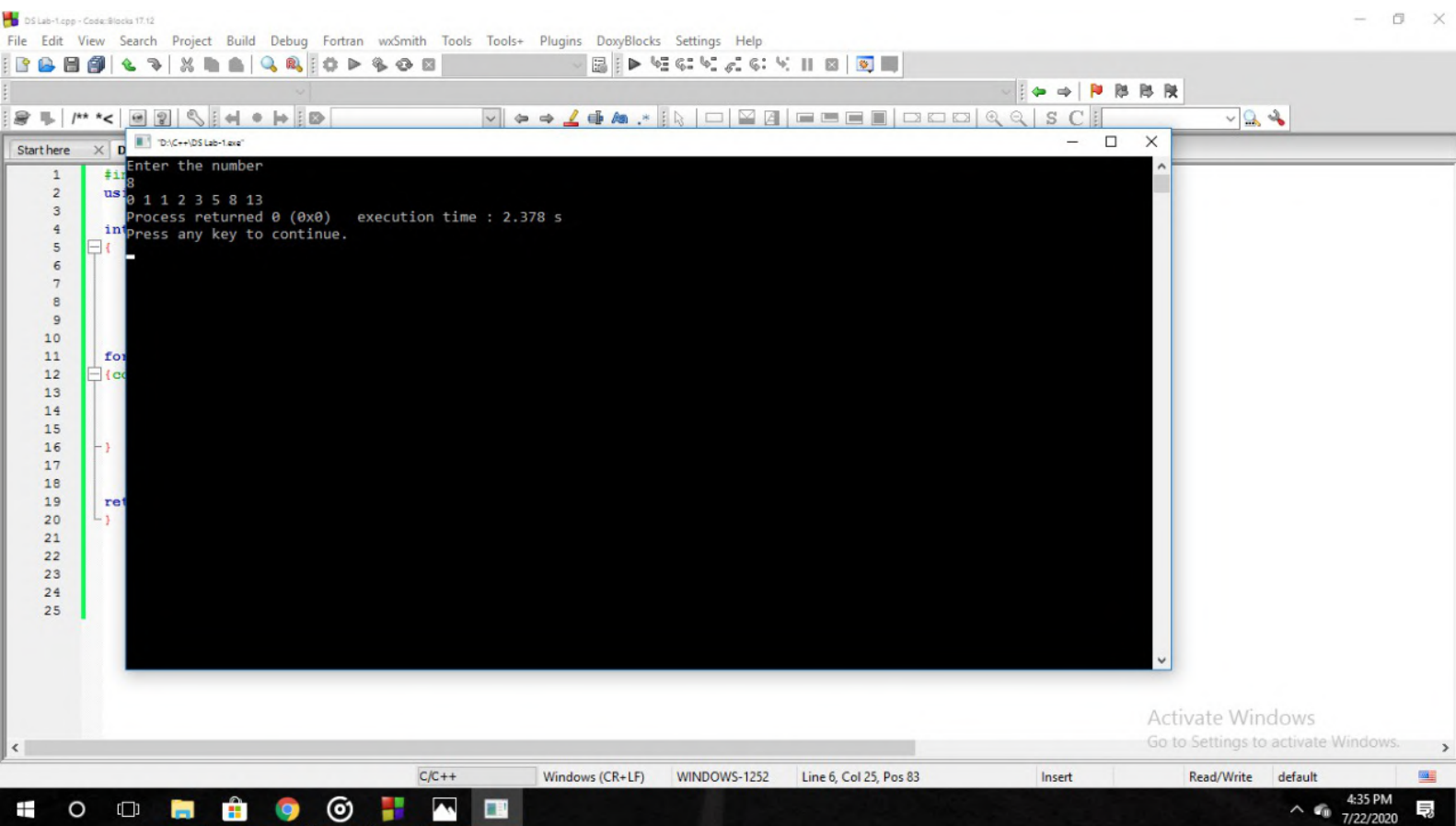
The screenshot shows a C++ code editor window titled "DS Lab-1.cpp" with the following code:

```
1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int n,s1=0,s2=1,i,s3; //space complexity=5 //time=2
7
8      cout<<"Enter the number"<<endl;
9      cin>>n;
10
11     for(i=1;i<=n;i++) //time=n
12     {cout<<s1<<" ";
13      s3=s1+s2; //time=1
14      s1=s2; //time=1
15      s2=s3; //time=1
16     }
17     //time complexity=2+3*n=3n+2=O(n).
18
19     return 0;
20 }
21
22
23
24
25
```

The code implements a Fibonacci series program. It prompts the user to enter a number 'n', then prints the first 'n' terms of the Fibonacci sequence. The space complexity is noted as 5 and the time complexity as  $O(n)$ .

At the bottom of the window, there is a status bar showing the file path "D:\C++\DS Lab-1.cpp", the language "C/C++", and the current position "Line 17, Col 35, Pos 313". The Windows taskbar at the very bottom shows the time as 4:37 PM on 7/22/2020.

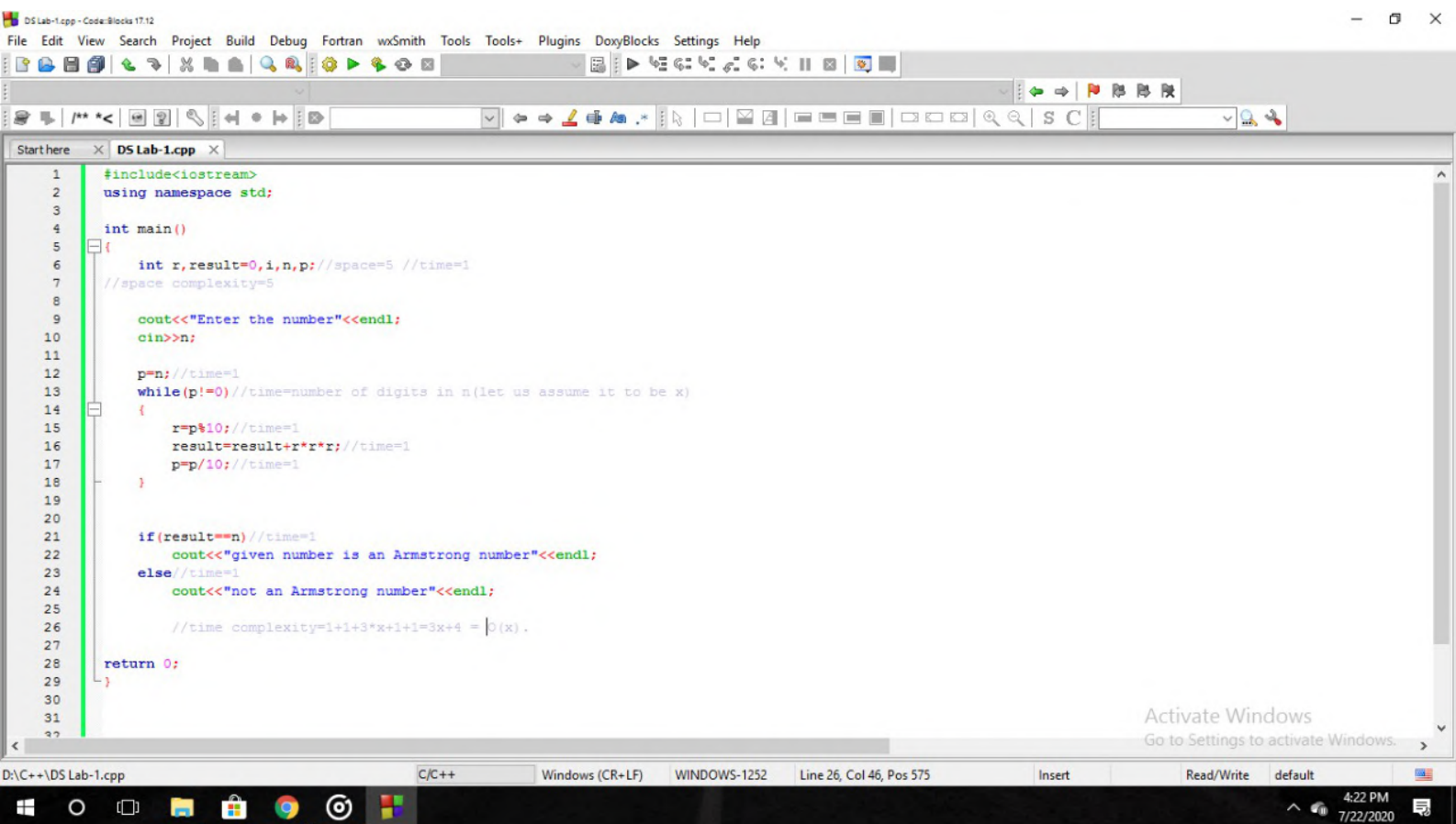
OUTPUT:



The screenshot shows a C++ IDE with a console window displaying the output of a program. The program prompts the user to "Enter the number" and then displays the input "0 1 1 2 3 5 8 13". It also shows the execution time as "2.378 s" and the message "Process returned 0 (0x0)". The console window is titled "D:\C++\IDS Lab-1.exe". The IDE interface includes a menu bar, a toolbar, and a status bar at the bottom. The status bar shows the current file is "C/C++", the window is "Windows (CR+LF)", the file size is "WINDOWS-1252", and the cursor is at "Line 6, Col 25, Pos 83". The system tray at the bottom right shows the time as "4:35 PM" and the date as "7/22/2020".

```
1  Enter the number
2  0 1 1 2 3 5 8 13
3  Process returned 0 (0x0)   execution time : 2.378 s
4  Press any key to continue.
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

## Q.2 WAP for Armstrong of number



The screenshot shows a C++ IDE with a single file named 'DS Lab-1.cpp'. The code is as follows:

```
1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int r,result=0,i,n,p;//space=5 //time=1
7      //space complexity=5
8
9      cout<<"Enter the number"<<endl;
10     cin>>n;
11
12     p=n;//time=1
13     while(p!=0)//time=number of digits in n(let us assume it to be x)
14     {
15         r=p%10;//time=1
16         result=result+r*r*r;//time=1
17         p=p/10;//time=1
18     }
19
20
21     if(result==n)//time=1
22         cout<<"given number is an Armstrong number"<<endl;
23     else//time=1
24         cout<<"not an Armstrong number"<<endl;
25
26     //time complexity=1+1+3*x+1+1=3x+4 = O(x).
27
28     return 0;
29 }
30
31
32
```

The IDE interface includes a menu bar (File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, Help), a toolbar, and a status bar at the bottom showing 'D:\C++\DS Lab-1.cpp', 'C/C++', 'Windows (CR+LF)', 'WINDOWS-1252', 'Line 26, Col 46, Pos 575', 'Insert', 'Read/Write', 'default', and the system clock '4:22 PM 7/22/2020'. An 'Activate Windows' watermark is visible in the bottom right corner.

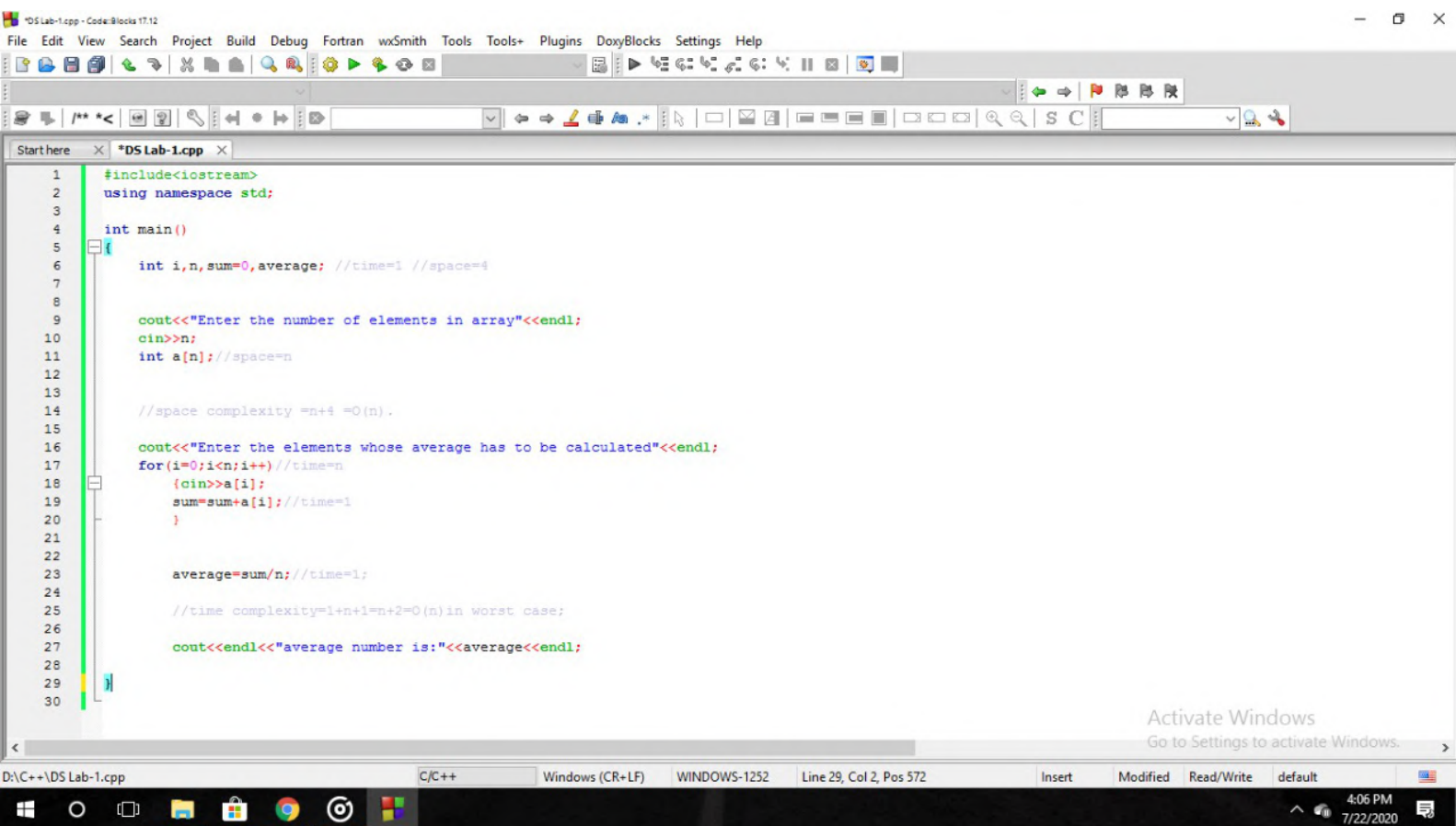
## OUTPUT:

```
D:\C++\DS Lab-1\ana
Enter the number
153
given number is an Armstrong number

Process returned 0 (0x0)   execution time : 6.084 s
Press any key to continue.
```

Activate Windows  
Go to Settings to activate Windows.

### Q.3 WAP for Average of number.



The screenshot shows a C++ code editor with the following code:

```
1  #include<iostream>
2  using namespace std;
3
4  int main()
5  {
6      int i,n,sum=0,average; //time=1 //space=4
7
8
9      cout<<"Enter the number of elements in array"<<endl;
10     cin>>n;
11     int a[n]; //space=n
12
13     //space complexity =n+4 =O(n).
14
15     cout<<"Enter the elements whose average has to be calculated"<<endl;
16     for(i=0;i<n;i++) //time=n
17     {
18         cin>>a[i];
19         sum=sum+a[i]; //time=1
20     }
21
22     average=sum/n; //time=1;
23
24     //time complexity=1+n+1=n+2=O(n) in worst case;
25
26     cout<<endl<<"average number is:"<<average<<endl;
27
28
29 }
30
```

The code is written in a C++ IDE with a menu bar (File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, Help) and a toolbar. The status bar at the bottom shows the file path 'D:\C++\DS Lab-1.cpp', the language 'C/C++', and the current position 'Line 29, Col 2, Pos 572'. The system tray at the bottom right shows the date and time '4:06 PM 7/22/2020'.



## OUTPUT:

```
D:\C++\DS\lab-1\ana
Enter the number of elements in array
5
Enter the elements whose average has to be calculated
2 5 7 14 50
average number is:15
Process returned 0 (0x0)   execution time : 10.273 s
Press any key to continue.
```

Activate Windows  
Go to Settings to activate Windows.

Data structure lab    second year

## Lab assignment 2

Name: Shalvi Singhal

Roll no: 0901CS191112

a) Gcd using simple school level algorithm

```
#include <iostream>
#include<time.h>
#include<math.h>
using namespace std;

void primeFactors(int n1,int n2,int min1)
{
    int gcd=1;
    while (n1 % 2 == 0&& n2%2==0)
    {
        gcd= gcd*2;
        n1 = n1/2;
        n2=n2/2;
    }

    for (int i = 3; i <= sqrt(min1); i = i + 2)
    {
        while (n1 % i == 0 && n2%i==0)
        {
            gcd=gcd*i;
            n1 = n1/i;
            n2 = n2/i;
        }
    }
}
```

```

    }
}

if (n1>2&& n2>2)

    {if(n1<n2)
        gcd=gcd*n1;
    else
        gcd=gcd*n2;
    }

    cout<< gcd<<endl;
}

int main()
{
    int n1, n2,gcd,min1;
    clock_t initial_time;

    cout << "Enter two numbers: ";
    cin >> n1 >> n2;

    initial_time=clock();

    if ( n2 > n1)
        min1=n1;
    else
        min1=n2;

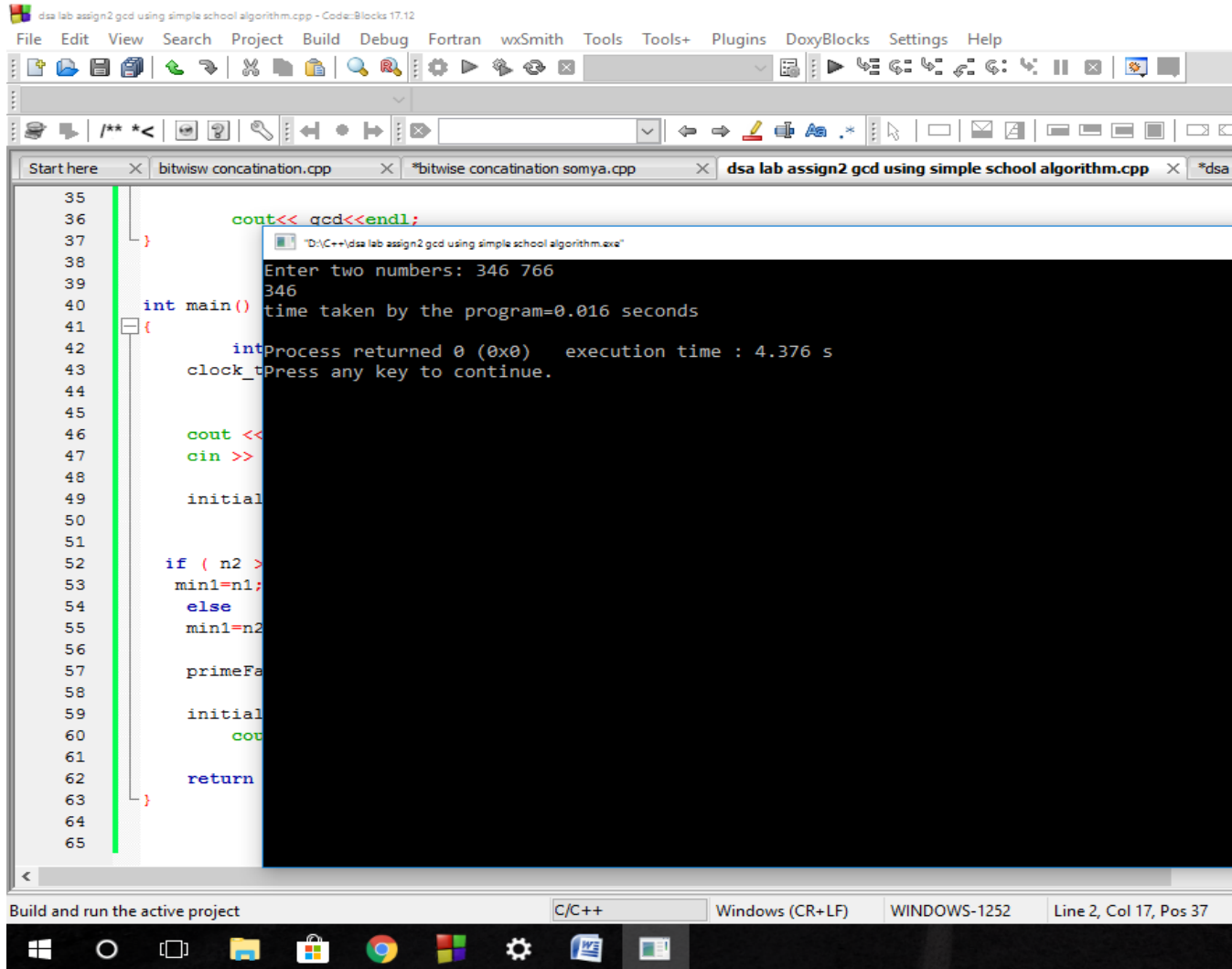
    primeFactors(n1,n2,min1);

    initial_time=clock()-initial_time;
    cout<<"time taken by the program="<<(float)initial_time/CLOCKS_PER_SEC<<"
seconds"<<endl;

    return 0;
}

```

OUTPUT:



The screenshot shows a C++ IDE with a code editor on the left and a console window on the right. The code editor displays a C++ program for finding the GCD of two numbers using a simple school algorithm. The console window shows the program's execution, including the input of two numbers (346 and 766) and the output of the GCD (346). The program also displays the execution time and a message to press any key to continue.

```
35  
36     cout<< gcd<<endl;  
37 }  
38  
39  
40 int main()  
41 {  
42     intProcess returned 0 (0x0)   execution time : 4.376 s  
43     clock_tPress any key to continue.  
44  
45  
46     cout <<  
47     cin >>  
48  
49     initial  
50  
51  
52     if ( n2 >  
53         min1=n1;  
54     else  
55         min1=n2  
56  
57     primeFa  
58  
59     initial  
60     cout  
61  
62     return  
63 }  
64  
65
```

Enter two numbers: 346 766  
346  
time taken by the program=0.016 seconds  
Process returned 0 (0x0) execution time : 4.376 s  
Press any key to continue.

Build and run the active project C/C++ Windows (CR+LF) WINDOWS-1252 Line 2, Col 17, Pos 37

## b) Gcd using Euclidean Algorithm

```
#include <iostream>
#include<time.h>
using namespace std;

int gcd(int a, int b)
{
    if (b == 0)
        return a;
    return gcd(b, a % b);
}

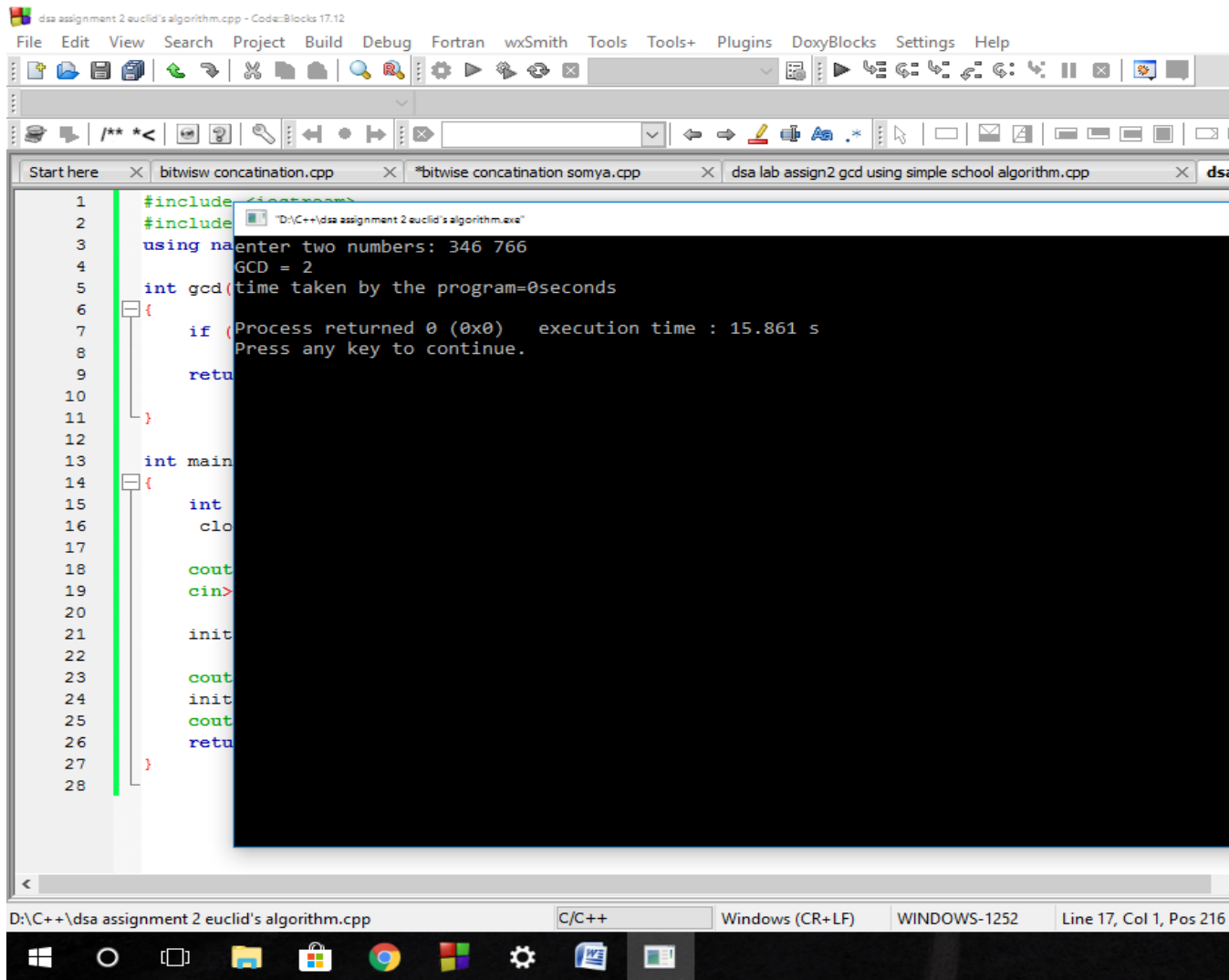
int main()
{
    int a,b;
    clock_t initial_time;

    cout<<"enter two numbers: ";
    cin>>a>>b;

    initial_time=clock();

    cout<<"GCD = "<<gcd(a, b)<<endl;
    initial_time=clock()-initial_time;
    cout<<"time taken by the
program="<<(float)initial_time/CLOCKS_PER_SEC<<"seconds"<<endl;
    return 0;
}
```

## OUTPUT:



The screenshot shows a C++ IDE with a code editor on the left and a console window on the right. The code editor displays a C++ program for calculating the GCD of two numbers. The console window shows the program's output, including the input numbers 346 and 766, the calculated GCD of 2, and the execution time of 15.861 seconds. The status bar at the bottom indicates the file path, compiler, encoding, and current line/col/pos.

```
1  #include <iostream>
2  #include "D:\C++\dsa assignment 2 euclid's algorithm.exe"
3  using namespace std;
4  GCD = 2;
5  int gcd(time taken by the program=0seconds
6  {
7      if (Process returned 0 (0x0)   execution time : 15.861 s
8          Press any key to continue.
9      return
10 }
11
12
13 int main
14 {
15     int
16     clo
17
18     cout
19     cin>
20
21     init
22
23     cout
24     init
25     cout
26     retu
27 }
28
```

Enter two numbers: 346 766  
GCD = 2  
time taken by the program=0seconds  
Process returned 0 (0x0) execution time : 15.861 s  
Press any key to continue.

D:\C++\dsa assignment 2 euclid's algorithm.cpp C/C++ Windows (CR+LF) WINDOWS-1252 Line 17, Col 1, Pos 216

# MADHAV INSTITUTE OF TECHNOLOGY AND SCIENCE



DS LAB

ASSIGNMENT III

Submitted to:  
Dr. Deepak Soni

Submitted by:  
Shalvi Singhal  
CSE-2<sup>nd</sup> year(III sem)  
0901CS191112

**Q1.** WAP to find largest number in an integer type array of size N where N and data elements are taken as input from the user.

```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int N,i,max1=INT_MIN;
    cout<<"Enter the size of the array : ";
    cin>>N;
    int a[N];

    cout<<"Enter the elements : "<<endl;
    for(i=0;i<N;i++)
    {
        cout<<"Enter "<<i+1<<"th element : ";
```

```

    cin>>a[i];
}

for(i=0;i<N;i++)
{
    if(a[i]>max1)
        max1=a[i];
}

cout<<"largest element is: "<<max1<<endl;
}

```

## Output:

```

ds lab assignment3 ques1.cpp - Code::Blocks 17.12
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
"D:\C++\ds lab assignment3 ques1.exe"
Enter the size of the array : 7
Enter the elements :
Enter 1th element : 4
Enter 2th element : 56
Enter 3th element : 44
Enter 4th element : 32
Enter 5th element : 1
Enter 6th element : 14
Enter 7th element : 35
largest element is: 56

Process returned 0 (0x0)   execution time : 14.626 s
Press any key to continue.
1
1
1
1
1
1
1
1
1
1
1
2
2
2
2
2
2
25
26

```

D:\C++\ds lab assignment3 ques1.cpp C/C++ Windows (CR+LF) WINDOWS-12



**Q2.** WAP to find sum of all the numbers stored in a float type array of size N, where N and data elements are taken as input from the user.

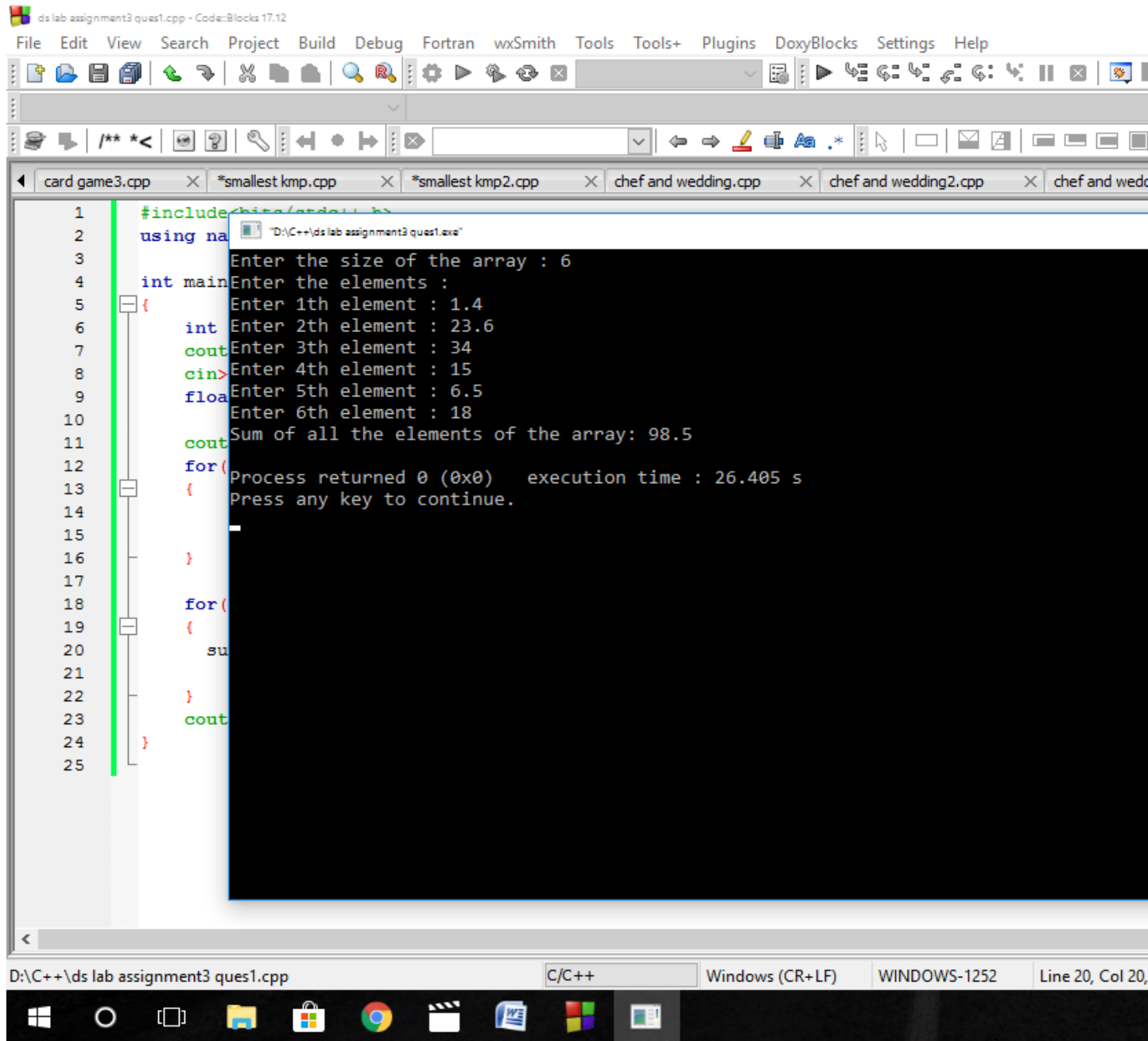
```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int N,i;
    cout<<"Enter the size of the array : ";
    cin>>N;
    float a[N],sum=0;

    cout<<"Enter the elements : "<<endl;
    for(i=0;i<N;i++)
    {
        cout<<"Enter "<<i+1<<"th element : ";
        cin>>a[i];
    }

    for(i=0;i<N;i++)
    {
        sum=sum+a[i];
    }
    cout<<"Sum of all the elements of the array: "<<sum<<endl;
}
```

## Output:



The screenshot shows a C++ IDE with a code editor on the left and a console window on the right. The code editor displays a C++ program that calculates the sum of an array. The console window shows the program's execution, including prompts for array size and elements, and the final sum.

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int n;
7      cout << "Enter the size of the array : ";
8      cin >> n;
9      float arr[n];
10     for (int i = 0; i < n; i++)
11     {
12         cout << "Enter the elements : ";
13         for (int j = 0; j < n; j++)
14         {
15             if (j == 0)
16                 cout << "Enter " << (i+1) << "th element : ";
17             else
18                 cout << "Enter " << (i+1) << "th element : ";
19             cin >> arr[j];
20         }
21     }
22     float sum = 0;
23     for (int i = 0; i < n; i++)
24     {
25         sum += arr[i];
26     }
27     cout << "Sum of all the elements of the array: " << sum << endl;
28     return 0;
29 }
```

Output:

```
Enter the size of the array : 6
Enter the elements :
Enter 1th element : 1.4
Enter 2th element : 23.6
Enter 3th element : 34
Enter 4th element : 15
Enter 5th element : 6.5
Enter 6th element : 18
Sum of all the elements of the array: 98.5
Process returned 0 (0x0)   execution time : 26.405 s
Press any key to continue.
```

**WAP** which takes **N** distinct integers as input in ascending order from the user and stores them in a dynamically created array of size **N**. Then it performs search operation for target value **X** entered by user according to the search technique chosen by user. if user press 1 than it performs linear search operation and if user press 2 than it performs binary search operation.

**Q3.** **Note:-** result of the search operation should display index number of the array at which target value **X** is present and number of comparisons it took to perform the search.

```
#include<bits/stdc++.h>
using namespace std;

int binarySearch(int arr[], int l, int g, int x)
{
    if (g>= l) {
        int mid = l + (g- l) / 2;

        if (arr[mid] == x)
            return mid;

        else if (arr[mid] > x)
            return binarySearch(arr, l, mid - 1, x);

        else if(arr[mid]<x)
            return binarySearch(arr, mid + 1, g, x);
    }

    return -1;
}

int main()
{
    int N,i,p,choice,flag=0;
    cout<<"Enter the number of elements you want to store: "<<endl;
    cin>>N;

    int *a=new int[N];

    cout<<"Enter "<<N<<" integers one by one in ascending order"<<endl;

    for(i=0;i<N;i++)
        {cout<<"Enter "<<i+1<<" th integer : ";
        cin>>a[i];
        }

    cout<<"Enter the element you want to search : "<<endl;
    cin>>p;

    cout<<"Press 1 for linear search"<<endl;
    cout<<"Press 2 for linear search"<<endl;
    cout<<"enter your choice : "<<endl;
    cin>>choice;
```

```

if(choice==1)
{
    for(i=0;i<N;i++)
    {
        if(a[i]==p)
        {cout<<"Element found at index no. : "<<i<<endl;
          cout<<"Number of comparisons : "<<i+1<<endl;
          flag=1;
          break;
        }
    }
    if(flag==0)
        cout<<"Element not found"<<endl;
}

else if(choice==2)
{int index;
  index=binarySearch(a,0,N,p);
  if(index!=-1)
  {cout<<"Element found at index no. : "<<index<<endl;

  int index1=N/2,index2=N/2,count1=1;
  while(index!=index1&&index!=index2)
  {
      index1=index1+(N-index1)/2;
      index2=index2-(index2-0)/2;
      count1++;
  }
  cout<<"Number of comparisons : "<<count1<<endl;
  }
  else
      cout<<"Element not found"<<endl;
}

else
    cout<<"Sorry invalid choice"<<endl;
}

```

**Output :**

"D:\C++\ds lab assignment3 ques3.exe"

Enter the number of elements you want to store:

10

Enter 10 integers one by one in ascending order

Enter 1 th integer : 10

Enter 2 th integer : 20

Enter 3 th integer : 30

Enter 4 th integer : 40

Enter 5 th integer : 50

Enter 6 th integer : 60

Enter 7 th integer : 70

Enter 8 th integer : 80

Enter 9 th integer : 90

Enter 10 th integer : 100

Enter the element you want to search :

1 80

1 Press 1 for linear search

1 Press 2 for linear search

1 Enter your choice :

1 2

1 Element found at index no. : 7

1 Number of comparisons : 2

1 Process returned 0 (0x0) execution time : 65.410 s

1 Press any key to continue.

1

2

2

2

2

2

2

2

2

2

2

2

2

2

30

31

32

`cout<<"Enter "<<N<<" integers one by one in ascending order"<<endl;`

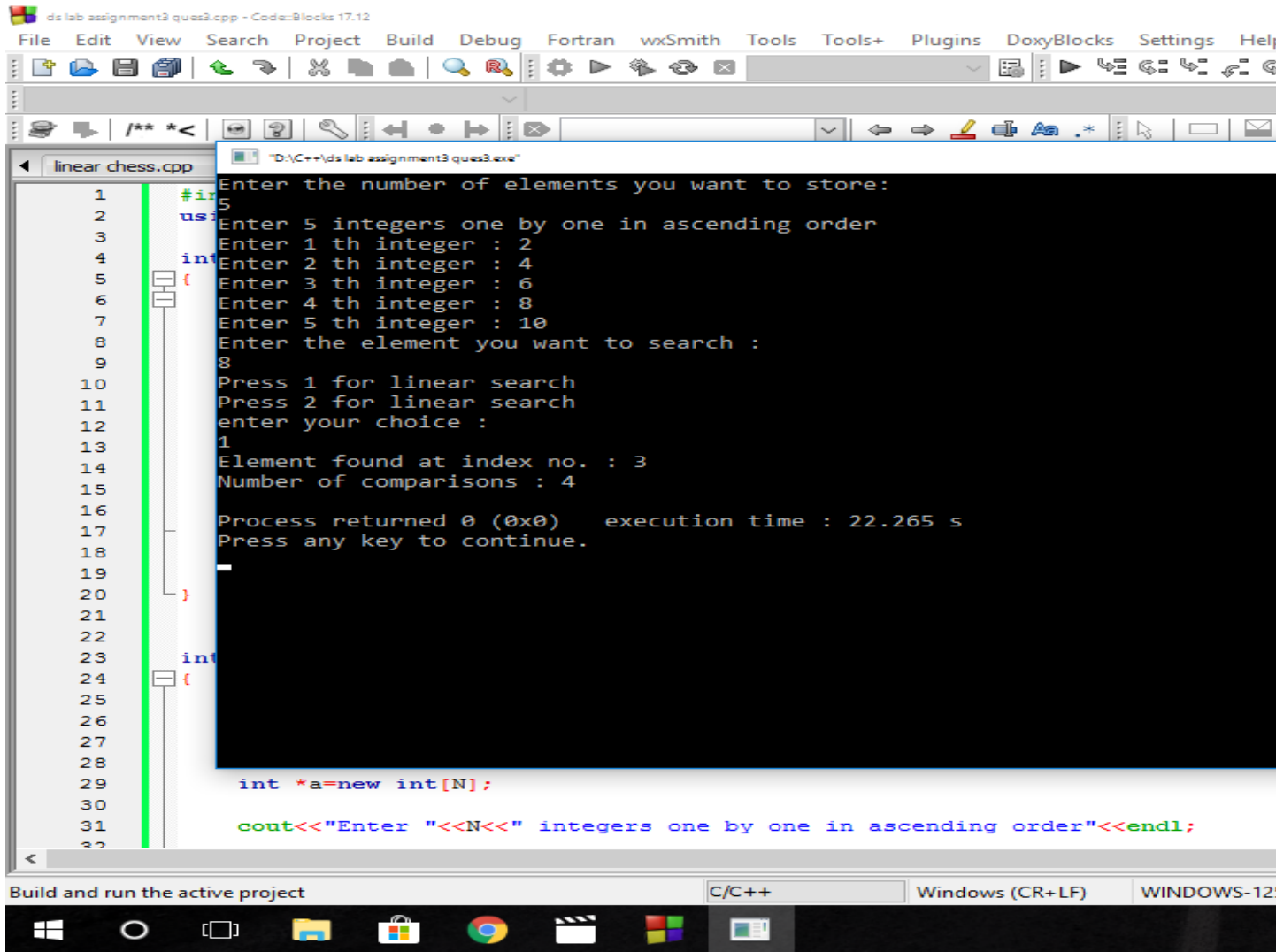
C/C++

Windows (CR+LF)

WINDOWS-1252

Lin





ds lab assignment3 ques3.cpp - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings

linear chess.cpp card game3.cpp \*smallest kmp.cpp \*smallest kmp2.cpp chef and wedding.cpp

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 int binarySearch(int arr[], int n, int x)
5 {
6     if (n < 1) return -1;
7     int low = 0, high = n - 1, mid;
8     while (low <= high)
9     {
10         mid = (low + high) / 2;
11         if (arr[mid] == x) return mid;
12         else if (arr[mid] < x) low = mid + 1;
13         else high = mid - 1;
14     }
15     return -1;
16 }
17
18 int main()
19 {
20     int N, i;
21     cout << "Enter the number of elements you want to store: ";
22     cin >> N;
23     int *a = new int[N];
24     cout << "Enter 7 integers one by one in ascending order\n";
25     for (i = 0; i < N; i++)
26     {
27         a[i] = i + 1;
28     }
29     int x;
30     cout << "Enter the element you want to search : ";
31     cin >> x;
32     int result = binarySearch(a, N, x);
33     if (result != -1)
34     {
35         cout << "Element found at index: " << result << endl;
36     }
37     else
38     {
39         cout << "Element not found\n";
40     }
41     return 0;
42 }
```

Process returned 0 (0x0) execution time : 22.382 s  
Press any key to continue.

D:\C++\ds lab assignment3 ques3.cpp C/C++ Windows (CR+LF) WINDOW

ds lab assignment3 ques3.cpp - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlo

Enter the number of elements you want to store:  
5  
Enter 5 integers one by one in ascending order  
Enter 1 th integer : 3  
Enter 2 th integer : 5  
Enter 3 th integer : 7  
Enter 4 th integer : 9  
Enter 5 th integer : 11  
Enter the element you want to search :  
5  
Press 1 for linear search  
Press 2 for linear search  
enter your choice :  
3  
Sorry invalid choice  
Process returned 0 (0x0) execution time : 13.800 s  
Press any key to continue.

linear che

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32

cout<<"Enter "<<N<<" integers one by one in ascending order

D:\C++\ds lab assignment3 ques3.cpp C/C++ Windows (CR+LF)



# MADHAV INSTITUTE OF TECHNOLOGY AND SCIENCE



DS LAB

ASSIGNMENT 5

Submitted to:  
Dr. Deepak Soni

Submitted by:  
Shalvi Singhal  
CSE-2<sup>nd</sup> year(III sem)  
0901CS191112

## Q1. WAP for reversing the element in array

```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int arr[50],i,j,N,temp;
    cout<<"Enter number of elements in the array : ";
    cin>>N;
    cout<<"Enter the elements : "<<endl;
    for(i=0;i<N;i++)
        cin>>arr[i];

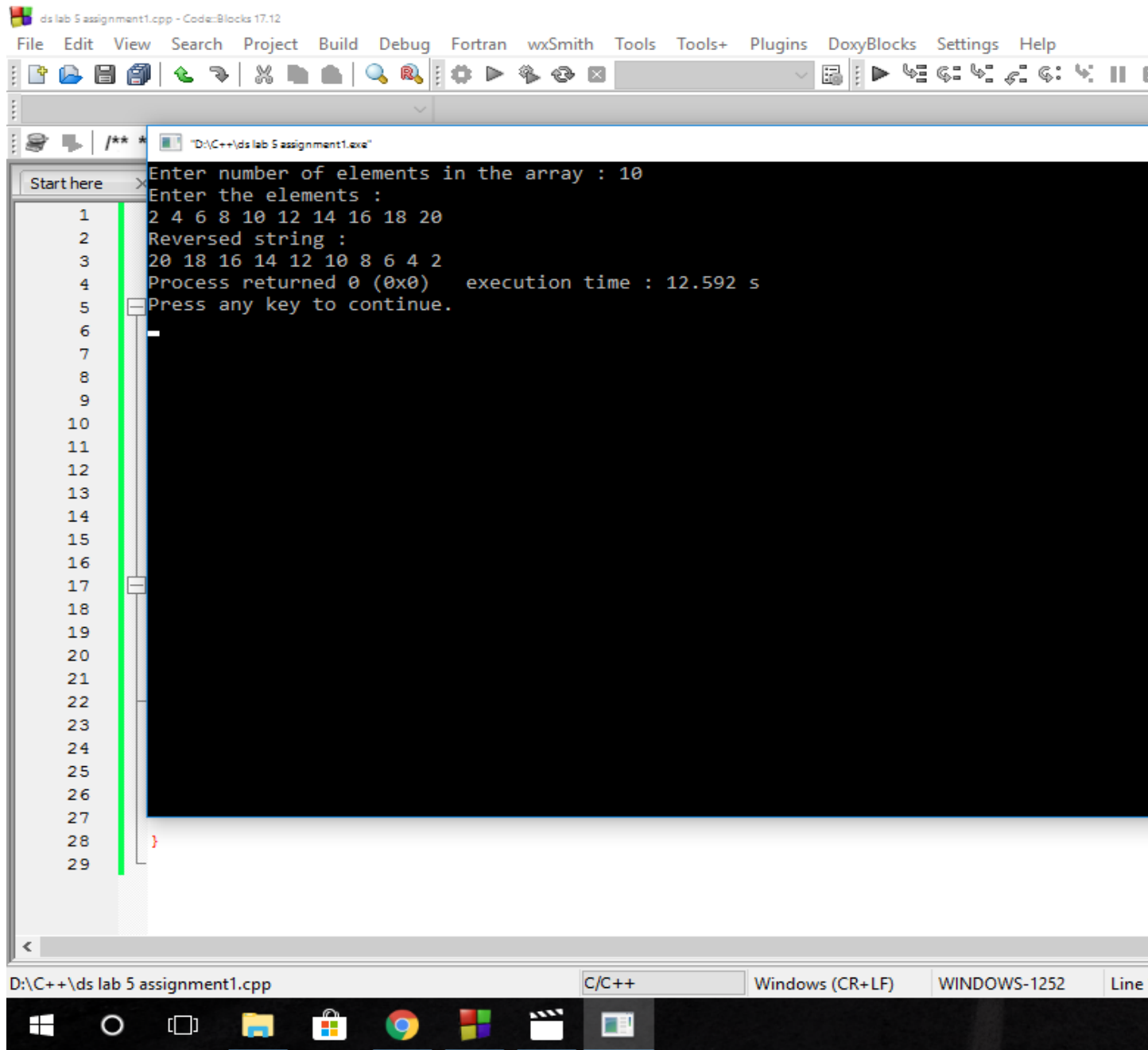
    i=0;
    j=N-1;

    while(i<j)
    {temp=arr[i];
    arr[i]=arr[j];
    arr[j]=temp;
    i++;
    j--;
    }

    cout<<"Reversed string : "<<endl;
    for(i=0;i<N;i++)
        cout<<arr[i]<<" ";

}
```

## OUTPUT:



The screenshot shows a C++ IDE with the following components:

- Menu Bar:** File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, Help.
- Toolbar:** Standard IDE icons for file operations, editing, and execution.
- Editor Window:** Titled "D:\C++\ds lab 5 assignment1.exe". It displays the output of a program:

```
Enter number of elements in the array : 10
Enter the elements :
2 4 6 8 10 12 14 16 18 20
Reversed string :
20 18 16 14 12 10 8 6 4 2
Process returned 0 (0x0)    execution time : 12.592 s
Press any key to continue.
```
- Source Code Window:** Titled "D:\C++\ds lab 5 assignment1.cpp". It shows a list of line numbers from 1 to 29. A green vertical line is positioned at line 17. The code is partially visible, showing a closing brace at line 28.
- Status Bar:** Displays "D:\C++\ds lab 5 assignment1.cpp", "C/C++", "Windows (CR+LF)", "WINDOWS-1252", and "Line".
- Taskbar:** Shows the Windows Start button and several open applications including File Explorer, Microsoft Store, Google Chrome, and the IDE.

## Q2. WAP for count odd and even numbers in array

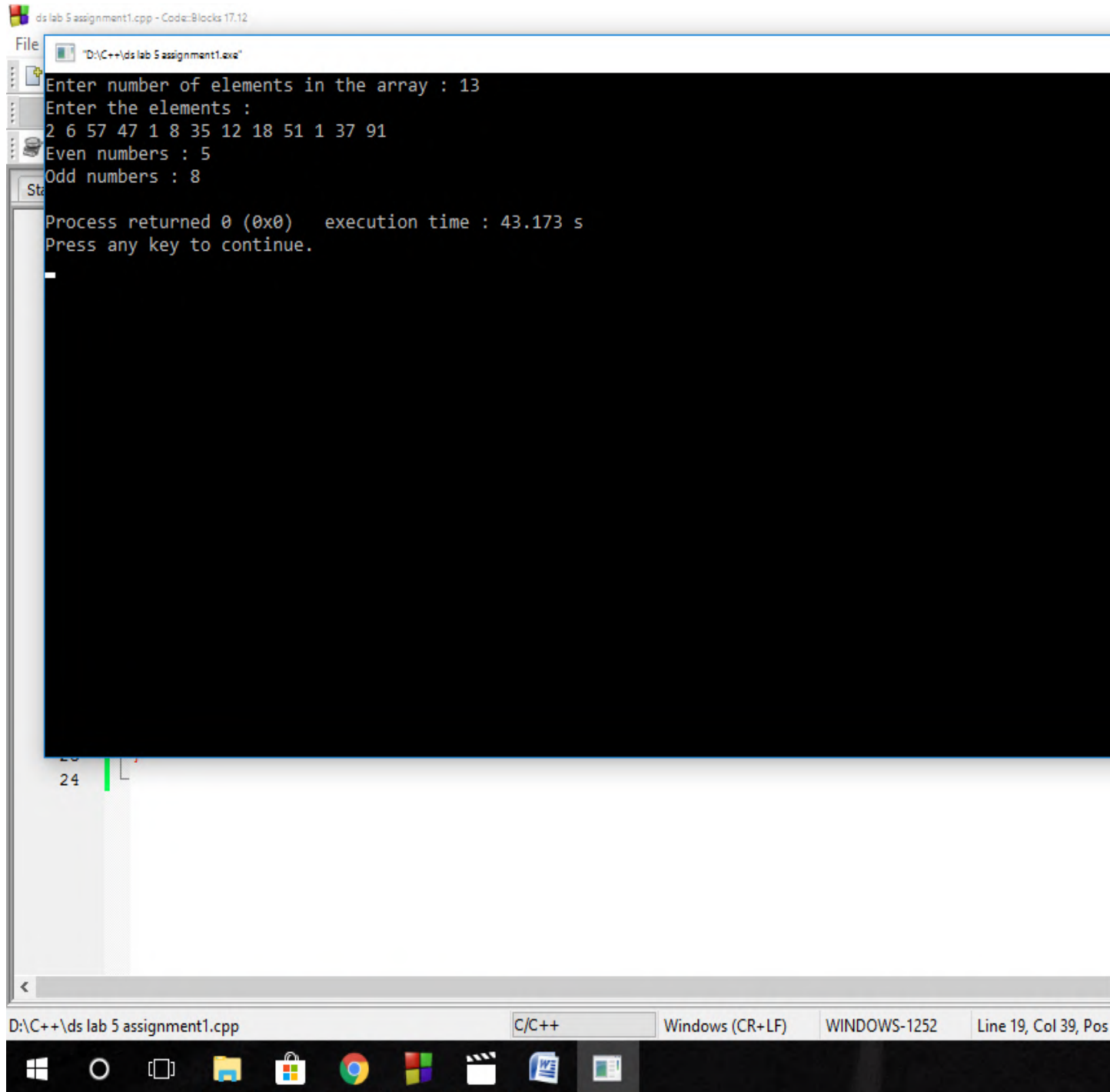
```
#include<bits/stdc++.h>
using namespace std;

int main()
{
    int arr[200],i,even=0,odd=0,N;
    cout<<"Enter number of elements in the array : ";
    cin>>N;
    cout<<"Enter the elements : "<<endl;
    for(i=0;i<N;i++)
    {
        cin>>arr[i];
        if(arr[i]%2==0)
            even++;
        else
            odd++;
    }

    cout<<"Even numbers : "<<even<<endl;
    cout<<"Odd numbers : "<<odd<<endl;

}
```

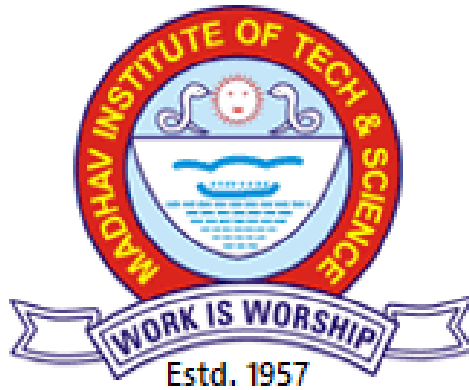
## OUTPUT:



```
ds lab 5 assignment1.cpp - Code::Blocks 17.12
File "D:\C++\ds lab 5 assignment1.exe"
Enter number of elements in the array : 13
Enter the elements :
2 6 57 47 1 8 35 12 18 51 1 37 91
Even numbers : 5
Odd numbers : 8
Process returned 0 (0x0)   execution time : 43.173 s
Press any key to continue.
```

D:\C++\ds lab 5 assignment1.cpp C/C++ Windows (CR+LF) WINDOWS-1252 Line 19, Col 39, Pos 24

# **MADHAV INSTITUTE OF TECHNOLOGY AND SCIENCE**



**DS LAB**

**Assignment – VII**

**Submitted To:  
Dr. Deepak Soni**

**Submitted By:  
Shalvi Singhal  
0901CS191112**

## Q. WAP to search an element in a Circular LinkedList

```
#include <bits/stdc++.h>
using namespace std;

class Node
{
public:
    int data;
    Node* next;
};

class circularLL
{
public:
    Node* head;
    circularLL()
    {
        head=NULL;
    }

    void create()
    {
        int N,x;
        cout<<"Enter the number of elements "<<endl;
        cin>>N;

        Node* p;
        cout<<"Enter the elements"<<endl;
        for(int i=0;i<N;i++)
        {
            cin>>x;
            if(head==NULL)
            {
                head=new Node;
                head->data=x;
                head->next=head;
                p=head;
            }
            else
            {
                Node* t=new Node;
```

```

        t->data=x;
        t->next=p->next;
        p->next=t;
        p=t;
    }

}

}

void display()
{
    Node* t=head;
    do
    {
        cout<<t->data<<" ";
        t=t->next;
    }
    while(t!=head);
    cout<<endl;
}

int searchElement(int key)
{
    int count=1;
    Node* t=head;
    do
    {
        if(t->data==key)return count;

        t=t->next;
        count++;
    }
    while(t!=head);

    return -1;
}

};

int main()
{
    circularLL c;
    int key;
    c.create();
    cout<<"the elements are : "<<endl;
    c.display();
    cout<<"Enter the value of key"<<endl;
    cin>>key;

```



```
int t=c.searchElement(key);
if(t>=0)
    cout<<"found at position : "<<t<<endl;
else
    cout<<"not found"<<endl;}
```

OUTPUT:

doubly linked list.cpp - Code::Blocks 17.12

```
File "D:\C++\doubly linked list.exe"
Enter the number of elements
5
Enter the elements
1 3 5 6 7
the elements are :
1 3 5 6 7
Enter the value of key
7
found at position : 5

Process returned 0 (0x0)   execution time : 4.749 s
Press any key to continue.
```

```
73         while(t!=head);
74
75         return -1;
76     }
77
78 };
79
80 int main()
81 {
```

D:\C++\doubly linked list.cpp

C/C++

Windows (CR+LF)

WIN





D:\C++\doubly linked list.exe

Start here

```
50 Enter the number of elements
51 5
52 Enter the elements
53 1 3 5 6 7
54 the elements are :
55 1 3 5 6 7
56 Enter the value of key
57 12
58 not found
59
60 Process returned 0 (0x0)   execution time : 9.212 s
61 Press any key to continue.
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78 }
79
80 int main()
81
```



## Q. WAP to search an element using doubly Linked list

```
#include <bits/stdc++.h>
using namespace std;

// Structure of a Node
struct Node
{
    int data;
    struct Node *next;
    struct Node *prev;
};

// Function to insert a node at the end
void insert(struct Node** head, int value)
{
    if (*head == NULL)
    {
        struct Node* p = new Node;
        p->data = value;
        p->next = p->prev = p;
        *head = p;
        return;
    }

    Node *tail = (*head)->prev;

    struct Node* p = new Node;
    p->data = value;

    p->next = *head;

    (*head)->prev = p;

    p->prev = tail;

    tail->next = p;
}

void display(struct Node* head)
```

```

{
    struct Node *t= head;

    while (t->next != head)
    {
        cout<< t->data<<" ";
        t = t->next;
    }
    cout<<t->data;
    cout<<endl;
}

```

```

int searchElement(struct Node* head, int key)
{

```

```

    struct Node *t = head;

```

```

    int count=0,flag=0;

```

```

    if(t == NULL)

```

```

        return -1;

```

```

    else

```

```

    {

```

```

        while(t->next != head)

```

```

        {

```

```

            count++;

```

```

            if(t->data == key)

```

```

            {

```

```

                flag = 1;

```

```

                count--;

```

```

                break;

```

```

            }

```

```

            t = t->next;

```

```

        }

```

```

    if(t->data == key)

```

```

    {

```

```

        count++;

```

```

        flag = 1;

```

```

    }

```

```

        if(flag == 1)
            cout<<endl<<key<<" found at position "<<
                count<<endl;
        else
            cout<<endl<<key<<" not found"<<endl;
    }
}

int main()
{int e;

    struct Node* head = NULL;


    insert(&head, 2);
    insert(&head, 4);
    insert(&head, 6);
    insert(&head, 8);
    insert(&head, 10);

    cout<<"Created doubly linked list is: ";
    display(head);
    cout<<"enter the element you want to search : "<<endl;
    cin>>e;
    searchElement(head, e);

    return 0;
}

```

OUTPUT:

 "D:\C++\doubly linked list.exe"

Created doubly linked list is: 2 4 6 8 10

enter the element you want to search :

6

6 found at position 3

Process returned 0 (0x0) execution time : 2.534 s

Press any key to continue.



"D:\C++\doubly linked list.exe"

Created doubly linked list is: 2 4 6 8 10

enter the element you want to search :

12

12 not found

Process returned 0 (0x0) execution time : 1.720 s

Press any key to continue.





# MADHAV INSTITUTE OF TECHNOLOGY AND SCIENCE



DS lab

ASSIGNMENT : 8

Submitted to:  
Prof. Deepak Soni

Submitted by:  
Shalvi Singhal  
CSE-2<sup>nd</sup> year(III sem)  
0901CS191112

## Stack implementation using linked list:

```
#include<bits/stdc++.h>
using namespace std;

class node
{public:
int data;
node *next;
};

class Stack
{
node *top;

public:
Stack(){top=NULL;}

void push(int x);
int pop();
void display();
};

void Stack::push(int x)
{
node *t=new node;
if(t==NULL)
cout<<"stack is full"<<endl;
else
{t->data=x;
t->next=top;
top=t;
}
}

int Stack::pop()
{
```

```

int x=-1;
if(top==NULL)
    cout<<"stack is empty"<<endl;
else
{
    x=top->data;
    node *t=top;
    top=top->next;
    delete t;
}
}

```

```

void Stack::display()
{
    node *p=top;
    while(p!=NULL)
    {
        cout<<p->data<<" ";
        p=p->next;
    }
    cout<<endl;
}

```

```

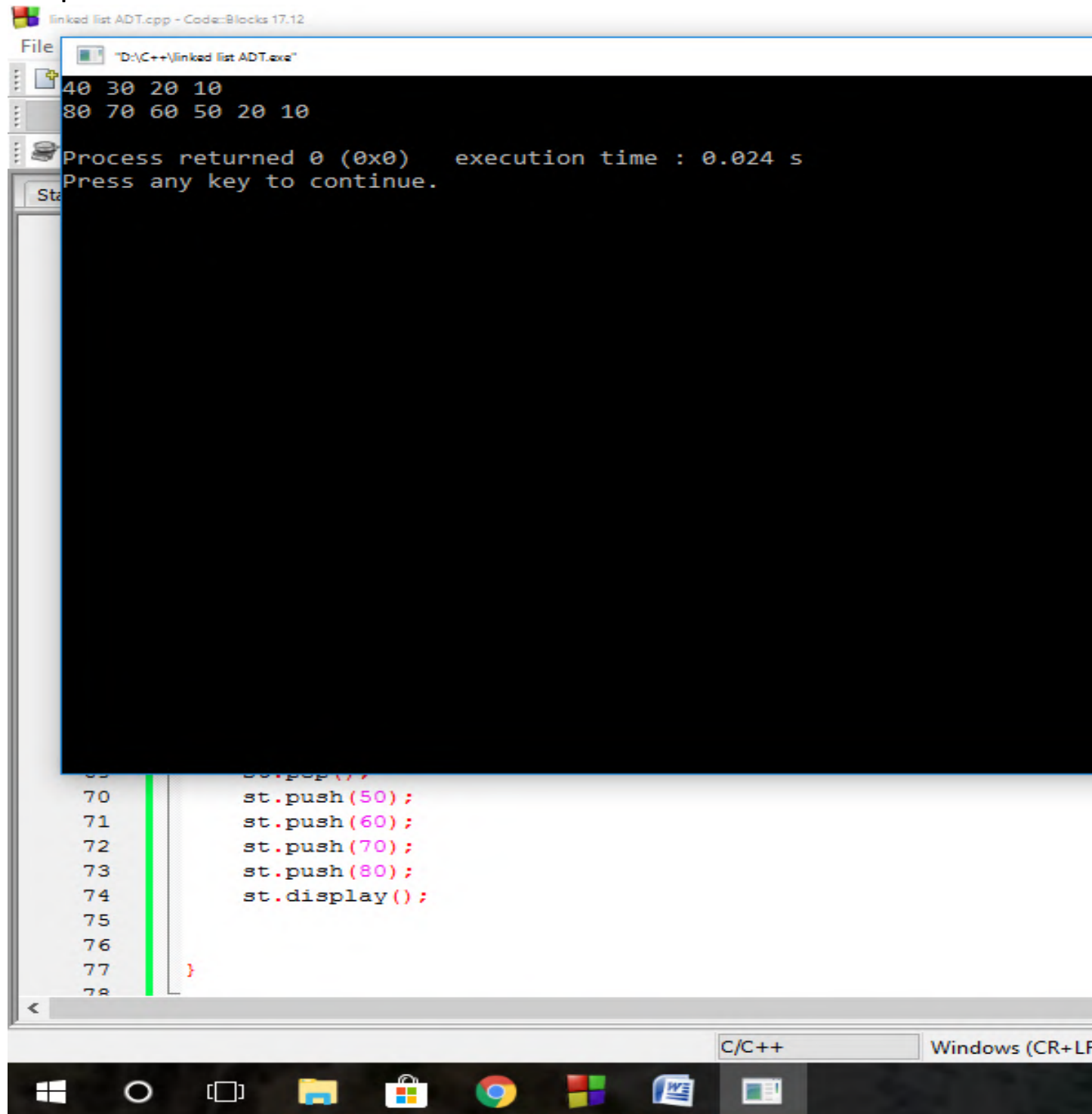
int main()
{
    Stack st;

    st.push(10);
    st.push(20);
    st.push(30);
    st.push(40);
    st.display();
    st.pop();
    st.pop();
    st.push(30);
    st.push(40);
    st.push(50);
}

```

```
    st.push(60);  
    st.display();  
}
```

Output:



The screenshot shows the Code::Blocks IDE with the following content:

**Console Window:**

```
40 30 20 10  
80 70 60 50 20 10  
Process returned 0 (0x0)   execution time : 0.024 s  
Press any key to continue.
```

**Source Code Window (linked\_list ADT.cpp):**

```
69  
70     st.push(50);  
71     st.push(60);  
72     st.push(70);  
73     st.push(80);  
74     st.display();  
75  
76  
77 }  
78
```

**Taskbar:** C/C++ Windows (CR+LF)

