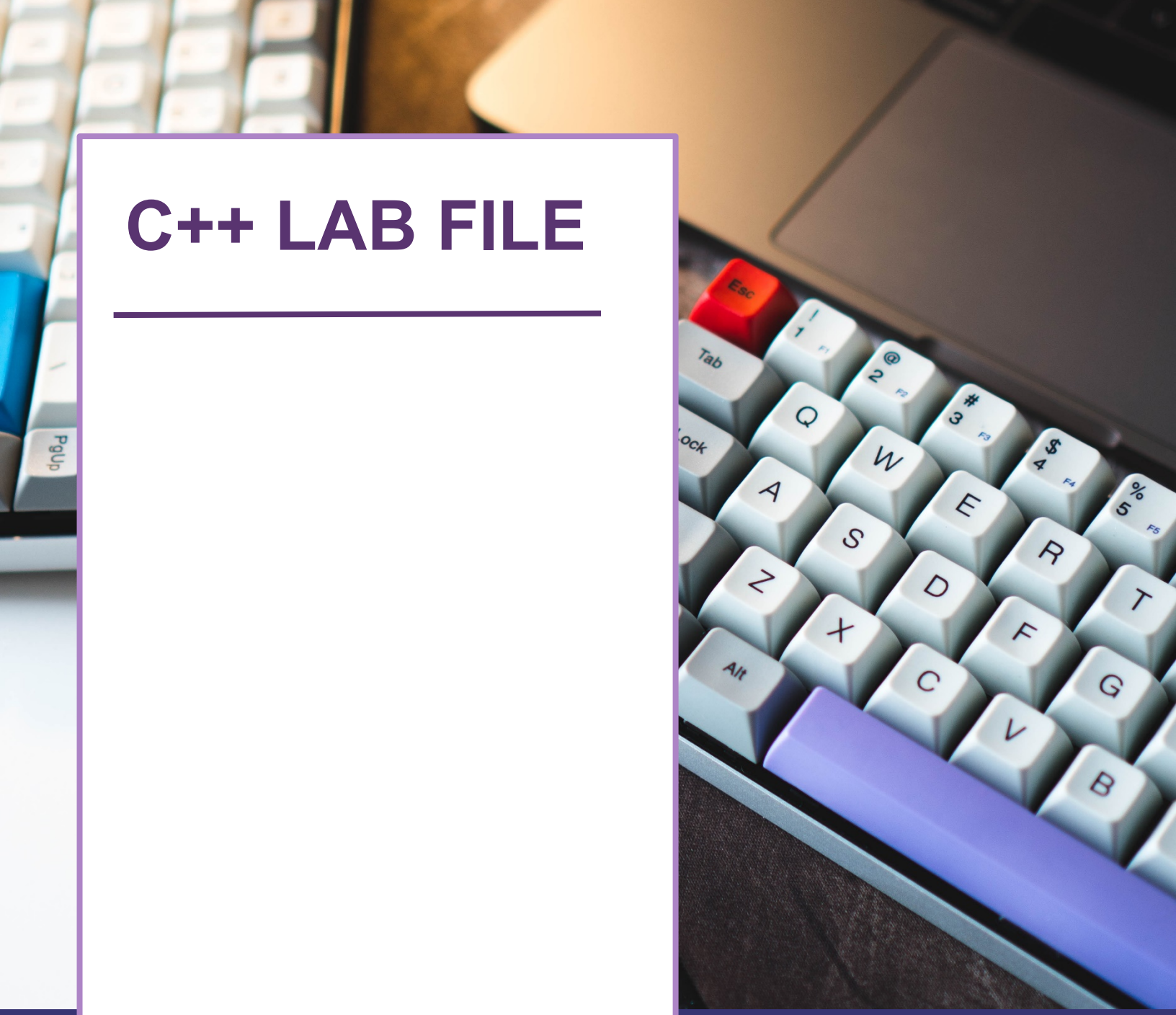


C++ LAB FILE

ATHARV KAVEESHWAR

Subject: C++ Programming
BSc.IT (AIIT)
Enrollment No: A710014918003
Roll no: BSIT1803
Submitted To: Mr. Sukhvinder Singh

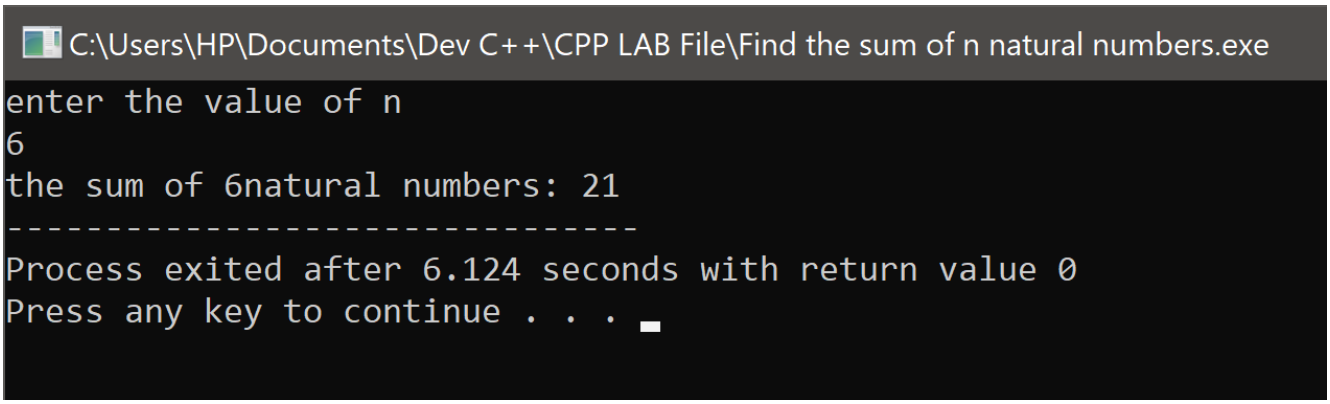
INDEX

1. Write a program using C++ Program

a. Find Sum of N natural numbers

```
#include<iostream>
#include<conio.h>
using namespace std;
main(){
int n, sum=0;
std::cout<<"enter the value of n"<<endl;
cin>>n;
for(int i=1; i<=n; i++){
sum=sum+i;}
cout<<"the sum of "<< n <<" natural numbers: "<<sum;
getch();
}
```

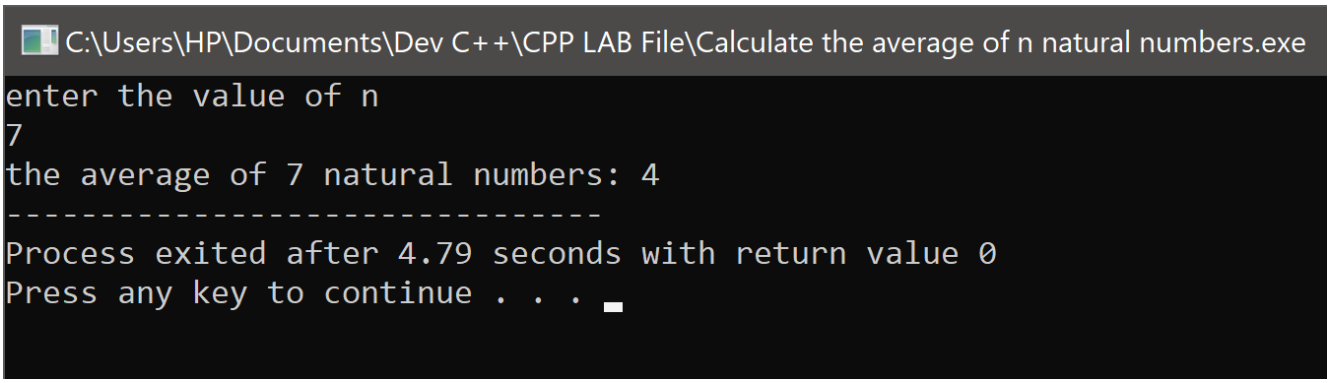
Output:



b. Find Average of N natural numbers

```
#include<iostream>
#include<conio.h>
using namespace std;
main(){
int n, sum=0;
std::cout<<"enter the value of n"<<endl;
cin>>n;
for(int i=1; i<=n; i++){
sum=sum+i;}
cout<<"the sum of "<< n <<" natural numbers: "<<sum;
getch();
}
```

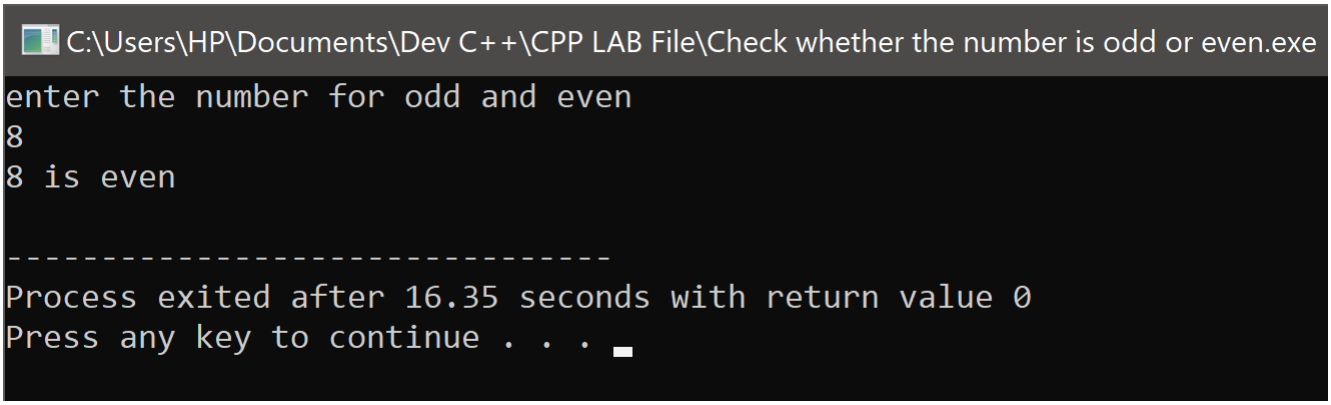
Output:



c. Find Odd or Even

```
1.  #include<iostream>
2.  #include<conio.h>
3.  using namespace std;
4.  main(){
5.  int n;
6.  cout<<"enter the number for odd and even"<<endl;
7.  cin>>n;
8.  if(n%2==0){
9.  cout<<n<<" is even"<<endl;
10. } else {
11. cout<<n<<" is odd"<<endl; }
12. getch();
13. }
```

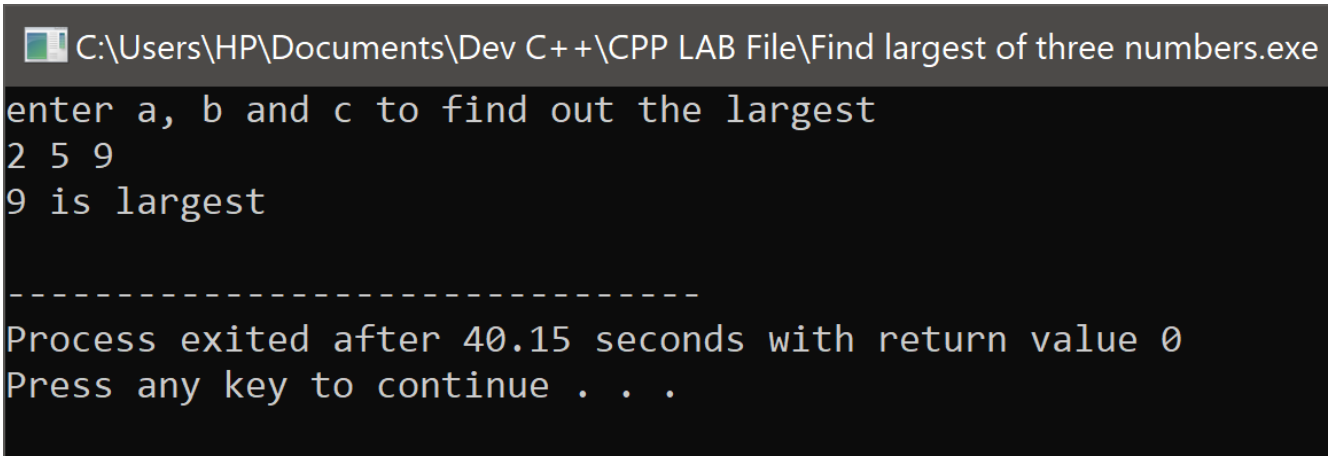
Output:



d. Find Largest of three numbers

```
#include<iostream>
#include<conio.h>
using namespace std;
main(){
int a,b,c;
cout<<"enter a, b and c to find out the largest"<<endl;
cin>>a>>b>>c;
if(a>b && a>c){
cout<<a<<" is largest"<<endl;
}
else if(b>a && b>c){
cout<<b<<" is largest"<<endl;
} else {
cout<<c<<" is largest"<<endl;
}
getch();
}
```

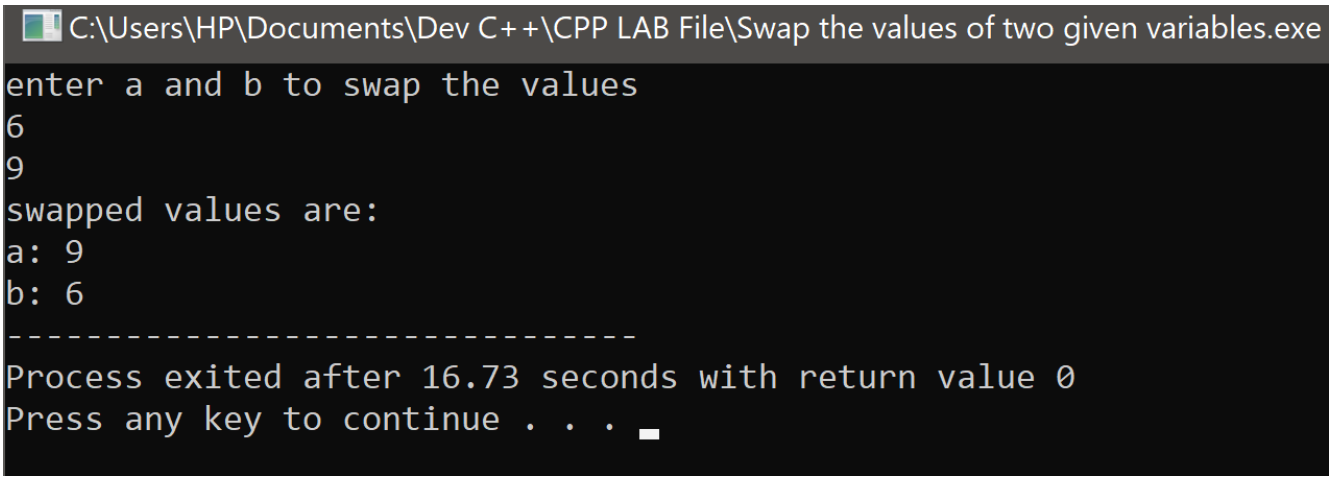
Output:



e. Swap the values

```
#include<iostream>
#include<conio.h>
using namespace std;
main(){
int a,b,c;
cout<<"enter a and b to swap the values"<<endl;
cin>>a>>b;
c=a;
a=b;
b=c;
cout<<"swapped values are:"<<endl;
cout<<"a: "<<a<<endl;
cout<<"b: "<<b;
getch();
}
```

Output:

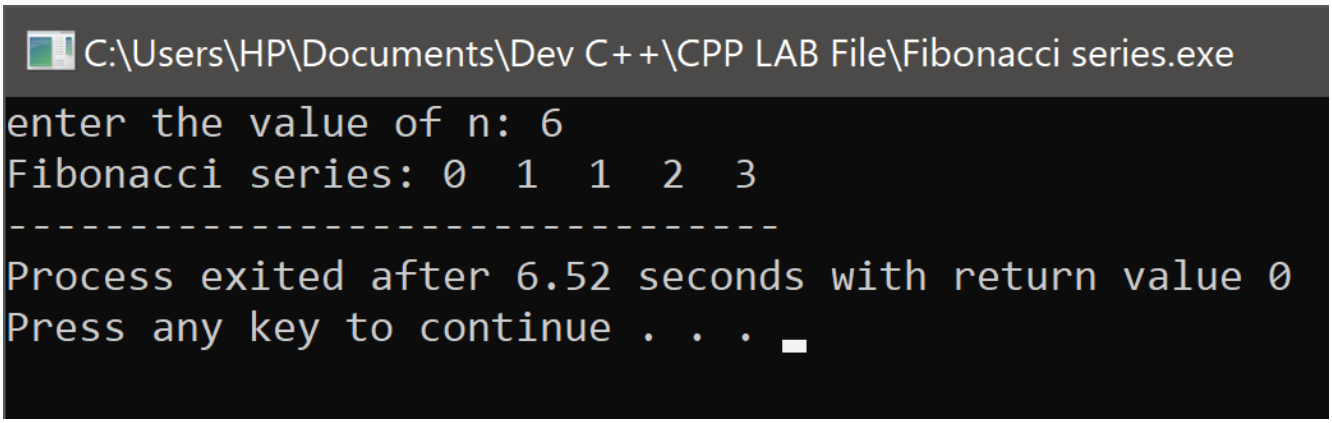


2. Write a Program using class, member data, member function and object.

a. Fibonacci Series

```
#include<iostream>
#include<conio.h>
using namespace std;
class fib{
int i, n,a,b,c;
public:
series(){
cout<<"enter the value of n: ";
cin>>n;
cout<<"Fibonacci series: ";
a=0; b=1; c=1;
cout<<a<<" ";
for(i=0; i<n-2; i++){
cout<<" "<<c<<" ";
c=a+b;
a=b;
b=c;
}
}
};
main(){
fib obj;
obj.series();
getch();
}
```

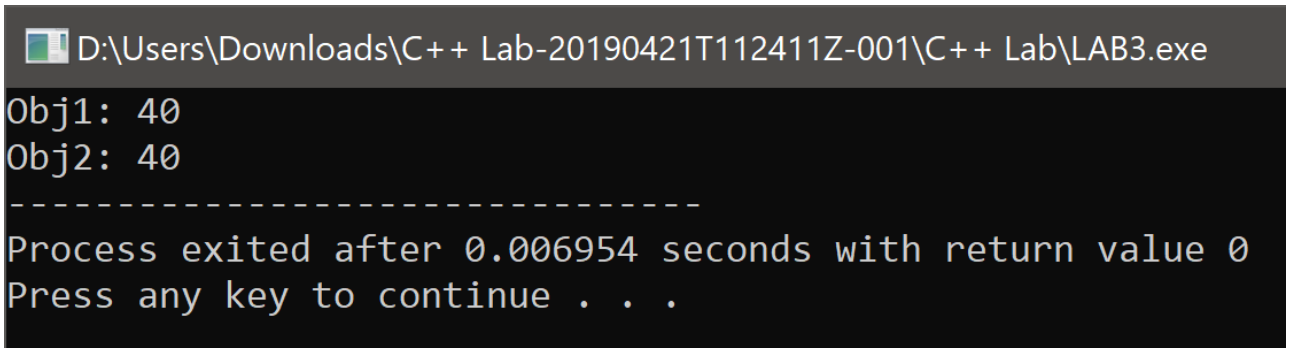
Output:



3. **Write a program using static variable to show its functionality. Show that static variable only access by the class and not belong to object.**

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

class StaticTest{
    private:
        static int s;

    public:
        void setValue(int num){
            s = num;
        }

        int getValue(){
            return s;
        }
};

int StaticTest::s = 0;

int main(){
    StaticTest obj1;
    obj1.setValue(20);  // sets value of s to 20

    StaticTest obj2;
    obj2.setValue(40);  // sets value of s to 40

    // value of s is changed for all objects
    cout << "Obj1: " << obj1.getValue() << endl;
    cout << "Obj2: " << obj2.getValue();
}
```

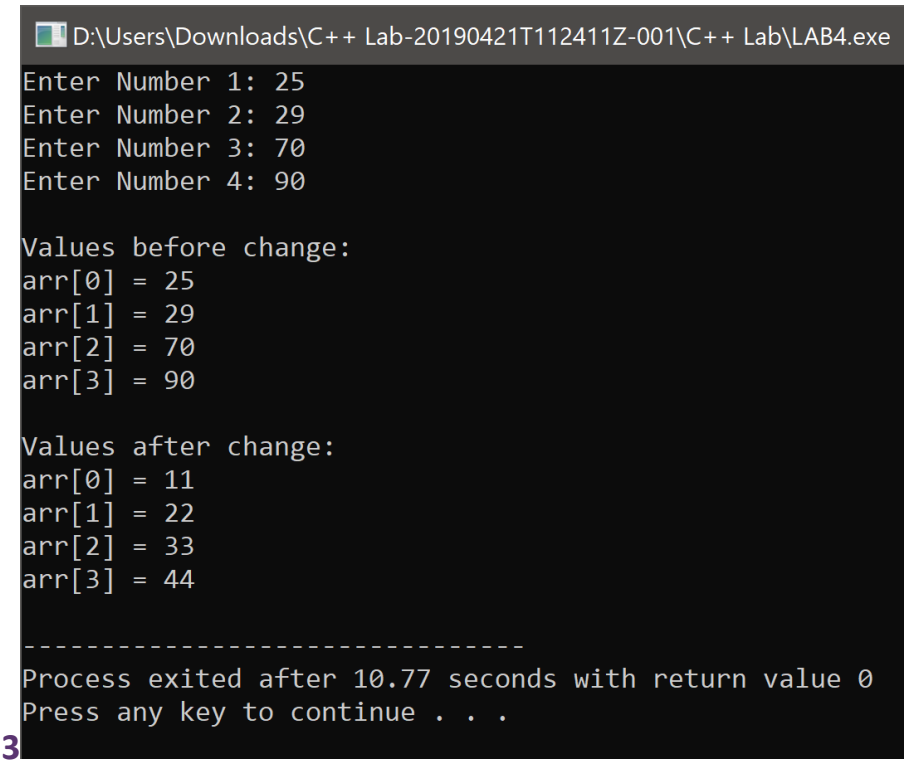
Output:



4. Write a program to return value by reference

```
#include<iostream>

using namespace std;

int arr[4], n = 4;

int& setValue(int index){
    return arr[index];
}

int main(){
    // taking user input values in array
    for(int i = 0; i < n; i++){
        cout << "Enter Number " << i + 1 << ": ";
        cin >> arr[i];
    }
    cout << endl;

    // values entered by user
    cout << "Values before change:" << endl;
    for(int i = 0; i < n; i++){
        cout << "arr" << "[" << i << "] = " << arr[i] << endl;
    }
    cout << endl;

    // changing values in array with return value by reference
    setValue(0) = 11;
    setValue(1) = 22;
    setValue(2) = 33;
    setValue(3) = 44;

    // values after they are modified
    cout << "Values after change:" << endl;
    for(int i = 0; i < n; i++){
        cout << "arr" << "[" << i << "] = " << arr[i] << endl;
    }
}
};
```

Output:

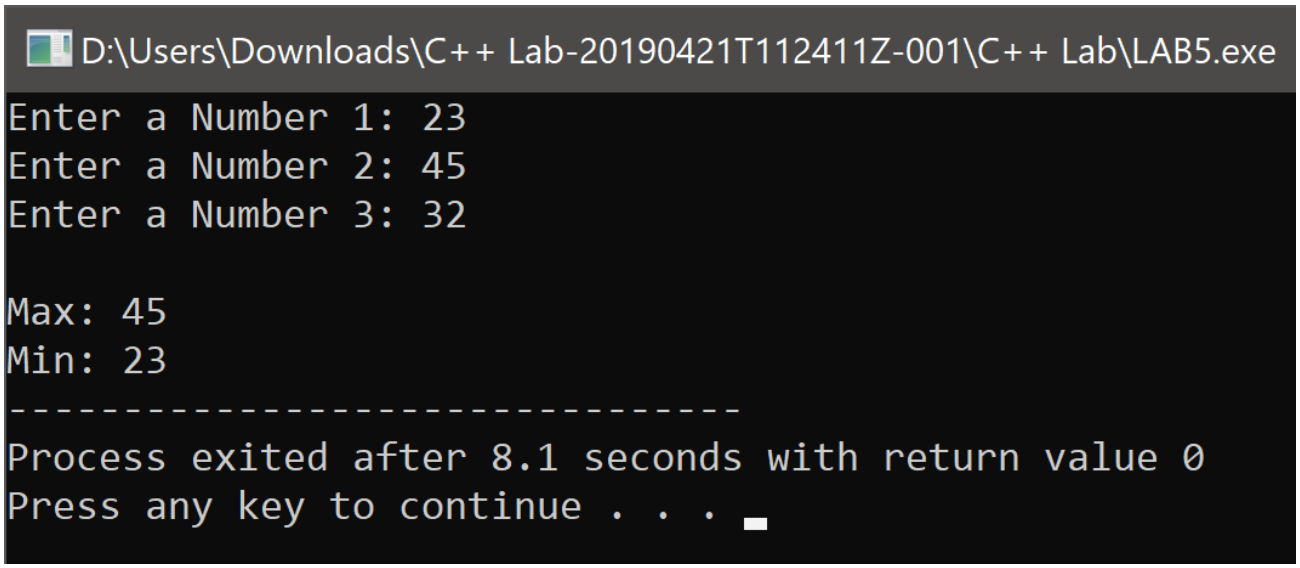


3

5. Write a program using inline function. Use any mathematical function to represent inline function.

```
#include<iostream>

using namespace std;

inline int max(int x, int y, int z){
    return (x > y) ? ((x > z) ? x : z) : ((y > z) ? y : z);
}

inline int min(int x, int y, int z){
    return (x < y) ? ((x < z) ? x : z) : ((y < z) ? y : z);
}

int main(){
    int a, b, c;

    cout << "Enter a Number 1: ";
    cin >> a;
    cout << "Enter a Number 2: ";
    cin >> b;
    cout << "Enter a Number 3: ";
    cin >> c;

    cout << endl;

    cout << "Max: " << max(a, b, c) << endl;
    cout << "Min: " << min(a, b, c); }
```

Output:



6. Write a Program using structures using pointers. Add five employees using structure using pointer.

```
1.  #include<iostream>
2.  #include<string>
3.  #include<sstream>
4.
5.  using namespace std;
6.
7.  struct Employee{
8.      int empId;
9.      string name;
10.     double salary;
11.     string dept;
12. };
13.
14. int main(){
15.     struct Employee emp[5];
16.     int n = 5;
17.     string s;
18.     struct Employee *ptr;
19.     ptr = emp;
20.
21.     cout << "ENTER EMPLOYEE DETAILS:" << endl;
22.     for(int i = 0; i < n; i++){
23.         cout << "-- Employee " << i + 1 << " --" << endl;
24.         cout << "ID: ";
25.         getline(cin, s);
26.         stringstream(s) >> emp[i].empId;
27.         cout << "Name: ";
28.         getline(cin, emp[i].name);
29.         cout << "Department: ";
30.         getline(cin, emp[i].dept);
31.         cout << "Salary: ";
32.         getline(cin, s);
33.         stringstream(s) >> emp[i].salary;
```

```
34.         cout << "----------------" << endl << endl;
35.     }
36.
37.     cout << endl << "YOU ENTERED:" << endl;
38.     for(int i = 0; i < n; i++){
39.         cout << "== Employee " << i + 1 << " ==" << endl;
40.         cout << "ID: " << (ptr+i)->empId << endl;
41.         cout << "Name: " << (ptr+i)->name << endl;
42.         cout << "Department: " << (ptr+i)->dept << endl;
43.         cout << "Salary: " << (ptr+i)->salary << endl;
44.         cout << "===============" << endl << endl;
45.     }
46. }
```

Output:

```
D:\Users\Downloads\C++ Lab-20190421T112411Z-001\C++ Lab\LAB6.exe
ENTER EMPLOYEE DETAILS:
-- Employee 1 --
ID: 01
Name: John Doe
Department: IT
Salary: 90000
---------------

-- Employee 2 --
ID: 02
Name: Jake Paul
Department: Social Media
Salary: 500000
---------------

-- Employee 3 --
ID: 03
Name: Sally Sane
Department: HR
Salary: 60000
---------------

-- Employee 4 --
ID: 04
Name: Emily Blunt
Department: Accounts
Salary: 75000
---------------

-- Employee 5 --
ID: 05
Name: Max Morgan
Department: Security
Salary: 62000
---------------
```

```
YOU ENTERED:
== Employee 1 ==
ID: 1
Name: John Doe
Department: IT
Salary: 90000
===============

== Employee 2 ==
ID: 2
Name: Jake Paul
Department: Social Media
Salary: 500000
===============

== Employee 3 ==
ID: 3
Name: Sally Sane
Department: HR
Salary: 60000
===============

== Employee 4 ==
ID: 4
Name: Emily Blunt
Department: Accounts
Salary: 75000
===============

== Employee 5 ==
ID: 5
Name: Max Morgan
Department: Security
Salary: 62000
===============
```

7. Write a Program using constructor and destructor.

```
1.  #include<iostream>
2.
3.  using namespace std;
4.
5.  class Car{
6.      private:
7.          int topSpeed;
8.
9.      public:
10.         Car(){
11.             cout << "Car created!" << endl;
12.             topSpeed = 250;
13.         }
14.
15.         Car(int ts){
16.             cout << "Car created with custom Top Speed!" << endl;
17.             topSpeed = ts;
18.         }
19.
20.         ~Car(){
21.             cout << "Car with Top Speed " << topSpeed << " destroyed!" << endl;
22.         }
23.
24.         int getTopSpeed(){
25.             return topSpeed;
26.         }
27. };
28.
29. int main(){
30.     Car C1;
31.     cout << "C1 Top Speed: " << C1.getTopSpeed() << endl << endl;
32.
33.     Car C2(500);
34.     cout << "C2 Top Speed: " << C2.getTopSpeed() << endl << endl;
35. }
```

Output:

```
D:\Users\Downloads\C++ Lab-20190421T112411Z-001\C++ Lab\LAB7.exe
Car created!
C1 Top Speed: 250

Car created with custom Top Speed!
C2 Top Speed: 500

Car with Top Speed 500 destroyed!
Car with Top Speed 250 destroyed!


--------------------------------
Process exited after 0.02028 seconds with return value 0
Press any key to continue . . .
```

8. Write a Program to show constructor and destructor functionality in inheritance.

```
#include<iostream>

using namespace std;

// base class
class Shape{
    protected:
        int id;
        int length;
        int height;

    public:
        Shape(){
            cout << "Shape created!" << endl;
        }

        Shape(int n){
            id = n;
            cout << "Shape (ID: " << id << ") created!" << endl;
        }

        ~Shape(){
            cout << "Shape (ID: " << id << ") destroyed!" << endl;
        }

        void setLength(int l){
            length = l;
        }

        void setHeight(int h){
            height = h;
        }
};

// derived class
class Rectangle: public Shape{
    public:
        Rectangle(int n){
            id = n;
            cout << "Rectangle (ID: " << id << ") created!" << endl;
        }

        ~Rectangle(){
            cout << "Rectangle (ID: " << id << ")  destroyed!" << endl;
        }

        int getArea(){
            return length * height;
        }
};

int main(){
    Shape s1(1);
    cout << endl;

    Rectangle r2(2);    // created rectangle with id 1
    r2.setLength(5);
```

```
58.     r2.setHeight(3);
59.     cout << endl;
60.
61.     Rectangle r3(3);    // created rectangle with id 2
62.     r3.setLength(9);
63.     r3.setHeight(7);
64.     cout << endl;
65.
66.     cout << "Rectangle 2 Area: " << r2.getArea() << endl << endl;
67.     cout << "Rectangle 3 Area: " << r3.getArea() << endl << endl;
68. }
```

Output:

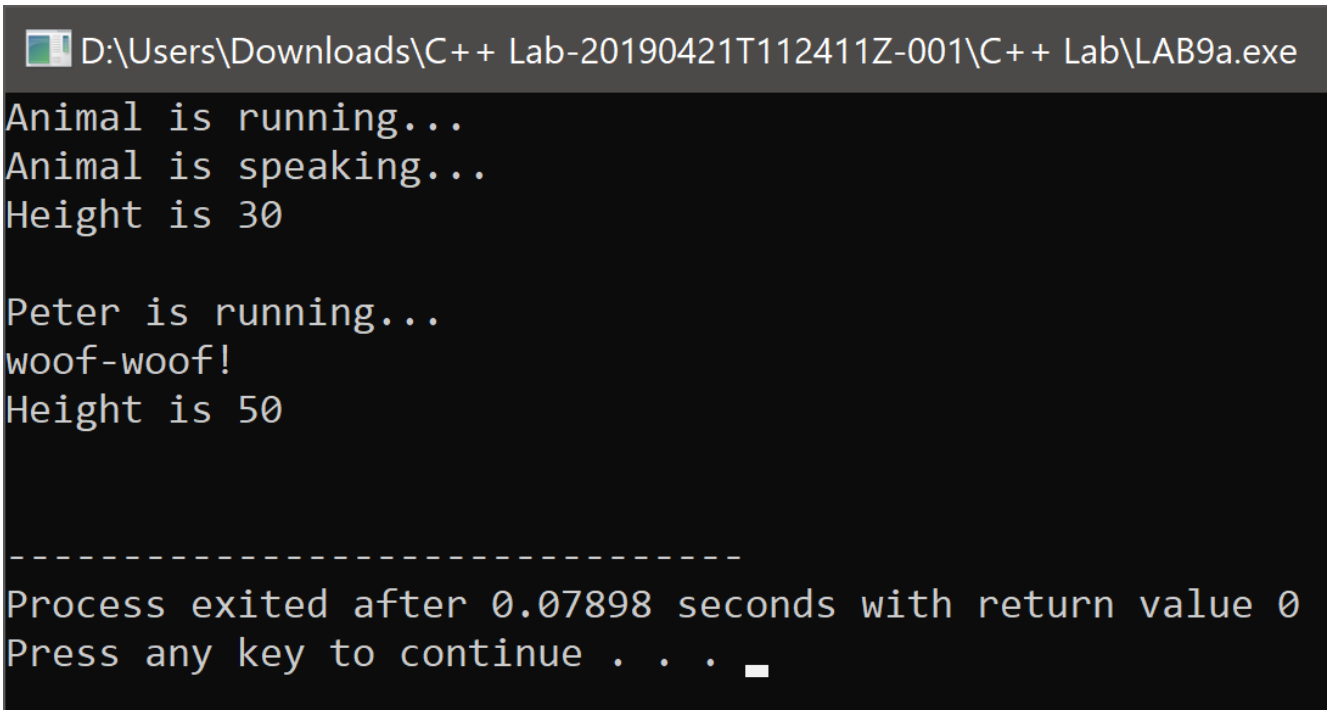
```
D:\Users\Downloads\C++ Lab-20190421T112411Z-001\C++ Lab\LAB8.exe
Shape (ID: 1) created!

Shape created!
Rectangle (ID: 2) created!

Shape created!
Rectangle (ID: 3) created!

Rectangle 2 Area: 15

Rectangle 3 Area: 63

Rectangle (ID: 3)  destroyed!
Shape (ID: 3) destroyed!
Rectangle (ID: 2)  destroyed!
Shape (ID: 2) destroyed!
Shape (ID: 1) destroyed!

--------------------------------
Process exited after 0.06262 seconds with return value 0
Press any key to continue . . .
```

9. Write a Program using Inheritance.

- **Single Inheritance**

```
#include<iostream>
#include<string>

using namespace std;

class Animal{
    protected:
        string name;
        int height;

    public:
        Animal(){
            name = "Animal";
        }

        void setHeight(int h){
            height = h;
        }

        void printHeight(){
            cout << "Height is " << height << endl;
        }

        void run(){
            cout << name << " is running..." << endl;
        }

        void speak(){
            cout << name << " is speaking..." << endl;
        }
};

class Dog: public Animal{
    public:
        Dog(string t){
            name = t;
            height = 50;
        }

        void speak(){
            cout << "woof-woof!" << endl;
        }
};

int main(){
    Animal a;
    a.run();
    a.speak();
    a.setHeight(30);
    a.printHeight();
    cout << endl;

    Dog d("Peter");
    d.run();
    d.speak();
    d.printHeight();
    cout << endl;  }
```

Output:

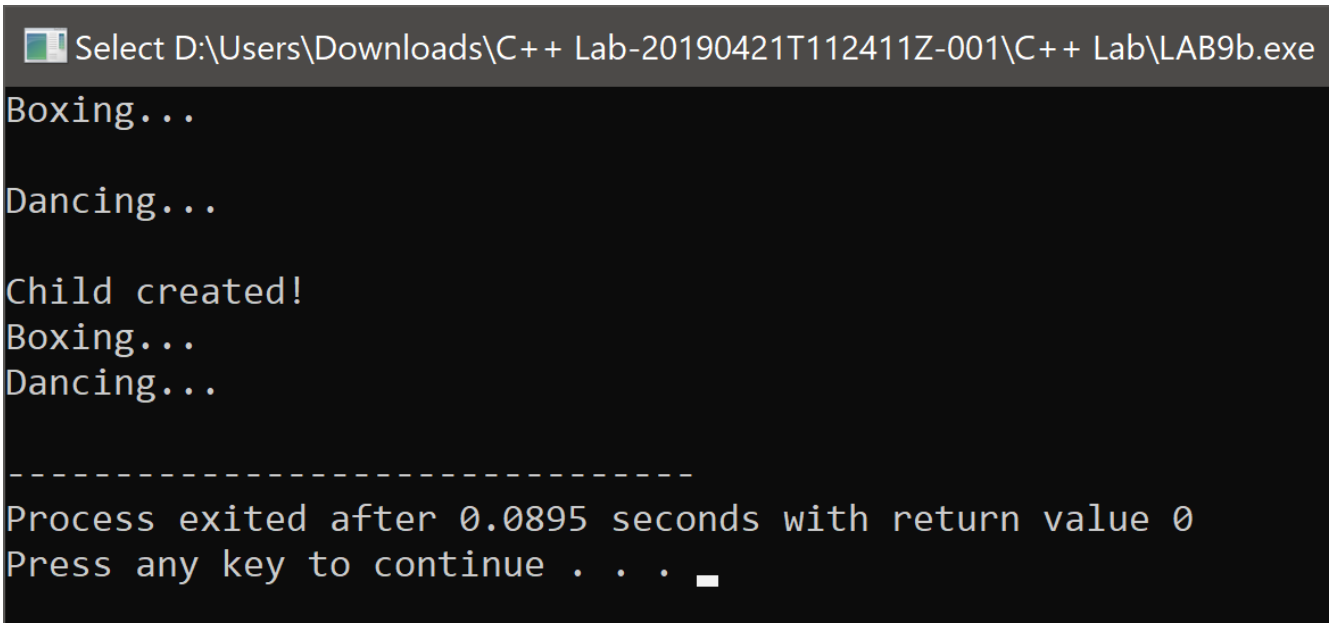


- **Multiple Inheritance**

```
1.  #include<iostream>
2.  #include<string>
3.
4.  using namespace std;
5.
6.  // base class
7.  class Father{
8.      public:
9.          void boxing(){
10.             cout << "Boxing..." << endl;
11.         }
12. };
13.
14. // base class
15. class Mother{
16.     public:
17.         void dance(){
18.             cout << "Dancing..." << endl;
19.         }
20. };
21.
22. // derived class
23. class Child: public Father, public Mother{
24.     public:
25.         Child(){
26.             cout << "Child created!" << endl;
27.         }
28. };
29.
30. int main(){
```

```
    Father f;
    f.boxing();

    cout << endl;

    Mother m;
    m.dance();

    cout << endl;

    // child inherits functions of both parents
    Child c;
    c.boxing();
    c.dance();
}
```

Output:

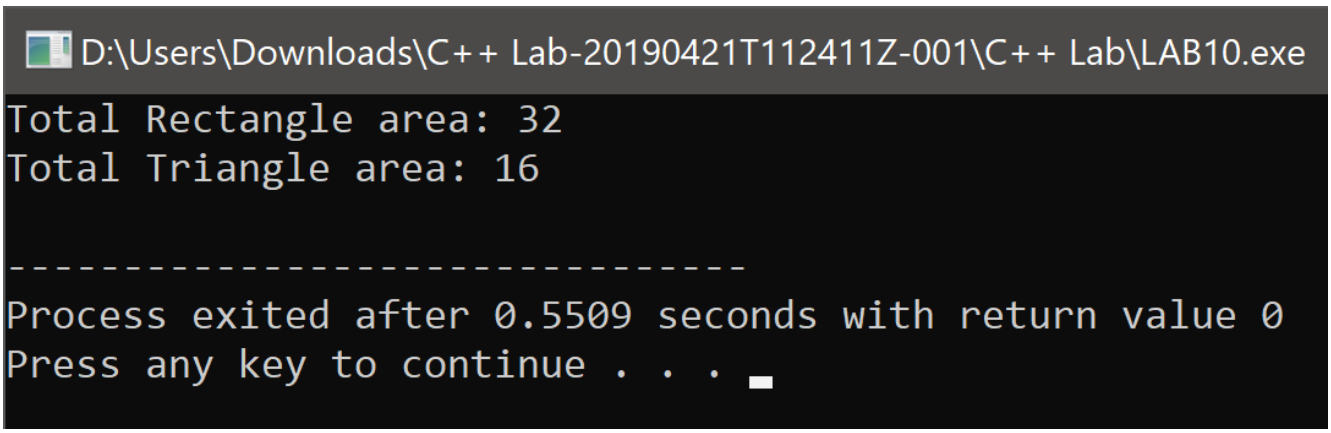


10. Write a program using abstract class and pure virtual function

```
#include<iostream>

using namespace std;

// base class
class Shape{
    protected:
        int width;
        int height;
    public:
        // pure virtual function providing interface framework.
        virtual int getArea() = 0;

        void setWidth(int w){
            width = w;
        }

```

```
18.         void setHeight(int h){
19.             height = h;
20.         }
21. };
22.
23. class Rectangle: public Shape{
24.     public:
25.         int getArea(){
26.             return width * height;
27.         }
28. };
29.
30. class Triangle: public Shape{
31.     public:
32.         int getArea(){
33.             return (width * height) / 2;
34.         }
35. };
36.
37. int main(){
38.     Rectangle rect;
39.     Triangle tri;
40.
41.     rect.setWidth(4);
42.     rect.setHeight(8);
43.
44.     // Prints area of the rectangle.
45.     cout << "Total Rectangle area: " << rect.getArea() << endl;
46.
47.     tri.setWidth(4);
48.     tri.setHeight(8);
49.
50.     // Prints area of the triangle.
51.     cout << "Total Triangle area: " << tri.getArea() << endl;
52. }
```

Output:



11. Write a Program to show diamond problem (ambiguity) and its resolution using virtual abstract class.

```
1.  #include<iostream>
2.
3.  using namespace std;
4.
5.  class A{
6.      public:
7.          void hello(){
8.              cout << "Hello from A!";
9.          }
10. };
11.
12. class B: virtual public A{
13.     public:
14. };
15.
16. class C: virtual public A{
17.     public:
18. };
19.
20. class D: public B, public C{
21.     public:
22. };
23.
24. int main(){
25.     D obj;
26.
27.     // classes B and C need to be made virtual, else hello() would be ambiguous
28.     obj.hello();
29. }
```

Output:

```
D:\Users\Downloads\C++ Lab-20190421T112411Z-001\C++ Lab\LAB11.exe
Hello from A!
--------------------------------
Process exited after 0.3059 seconds with return value 0
Press any key to continue . . .
```

12. Write a Program to write and read file using stream in C++.

```cpp
#include<iostream>
#include<fstream>

using namespace std;

int main(){
    // writing to file
    char data[100];

    ofstream outfile;
    outfile.open("A12.txt", ios::out);

    cout << "== Writing to file ==" << endl;
    cout << "Enter your name: ";
    cin.getline(data, 100);
    outfile << data << endl;

    outfile.close();

    // reading from file
    ifstream infile;
    infile.open("A12.txt", ios::in);

    cout << endl << "== Reading from file ==" << endl;
    infile >> data;

    cout << "Your name: " << data << endl;

    infile.close();
}
```

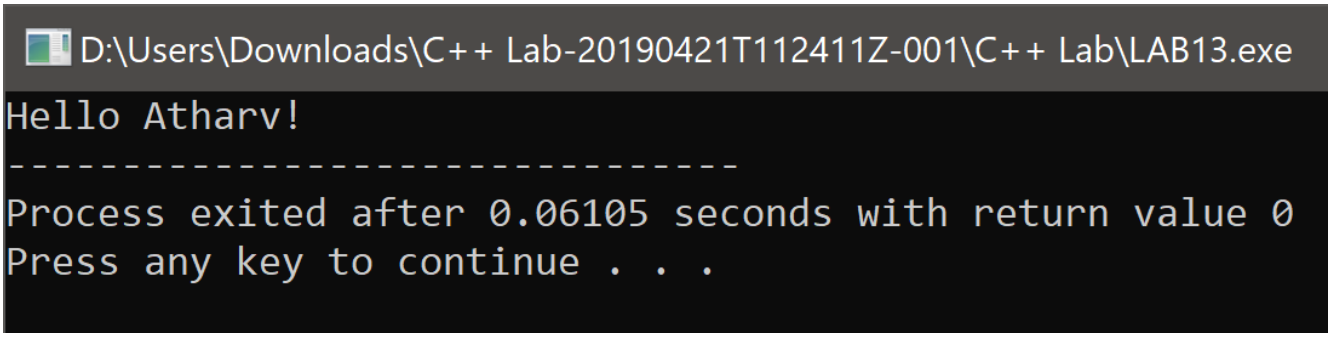
Output:

```
D:\Users\Downloads\C++ Lab-20190421T112411Z-001\C++ Lab\LAB12.exe
== Writing to file ==
Enter your name: Atharv

== Reading from file ==
Your name: Atharv

--------------------------------
Process exited after 5.079 seconds with return value 0
Press any key to continue . . .
```

13. Write a Program using command line arguments.

```
#include<iostream>

using namespace std;

int main(int argc, char* argv[]){
    if(argc != 2){
        cout << "You forgot to type your name.";
        return 1;
    }

    cout << "Hello " << argv[1] << "!";
}
```

Output:

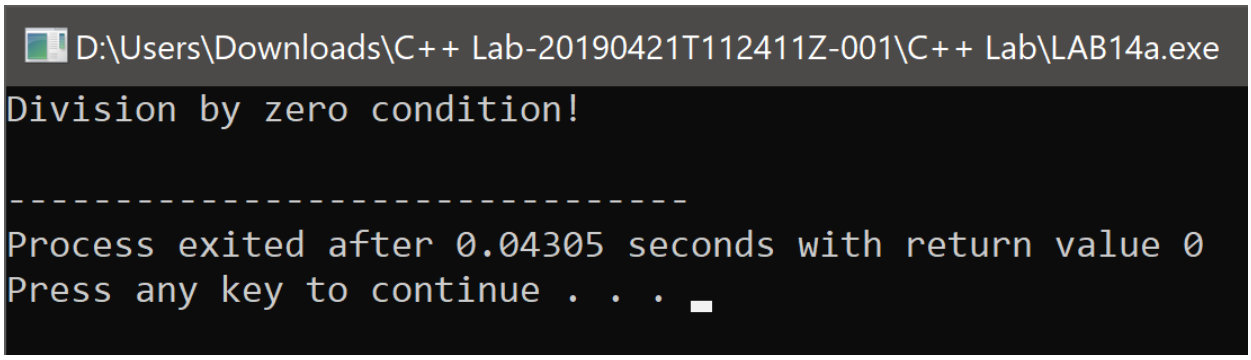


14. WAP on Exception Handling

- **Division by zero exception**

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

double division(int a, int b) {
   if( b == 0 ) {
      throw "Division by zero condition!";
   }
   return (a/b);
}

int main () {
   int x = 50;
   int y = 0;
   double z = 0;

   try {
      z = division(x, y);
      cout << z << endl;
   } catch (const char* msg) {
     cerr << msg << endl;
   }
   return 0;  }
```

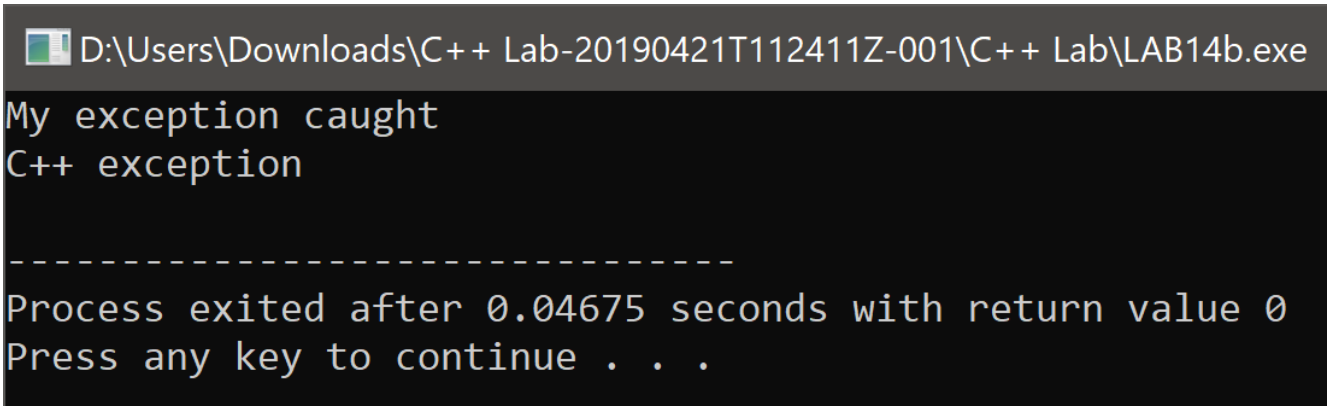
Output:



- **Using Std::exception class to implement your own exception**

```
1.  #include<iostream>
2.  #include<exception>
3.  using namespace std;
4.
5.  struct myException: public exception
6.  {
7.      const char *what() const throw()
8.      {
9.          return "C++ exception";
10.     }
11. };
12.
13. int main()
14. {
15.     try
16.     {
17.         throw myException();
18.     }
19.     catch(myException &e)
20.     {
21.         std::cout<<"My exception caught"<<std::endl;
22.         std::cout<<e.what()<<std::endl;
23.     }
24. }
```

Output:

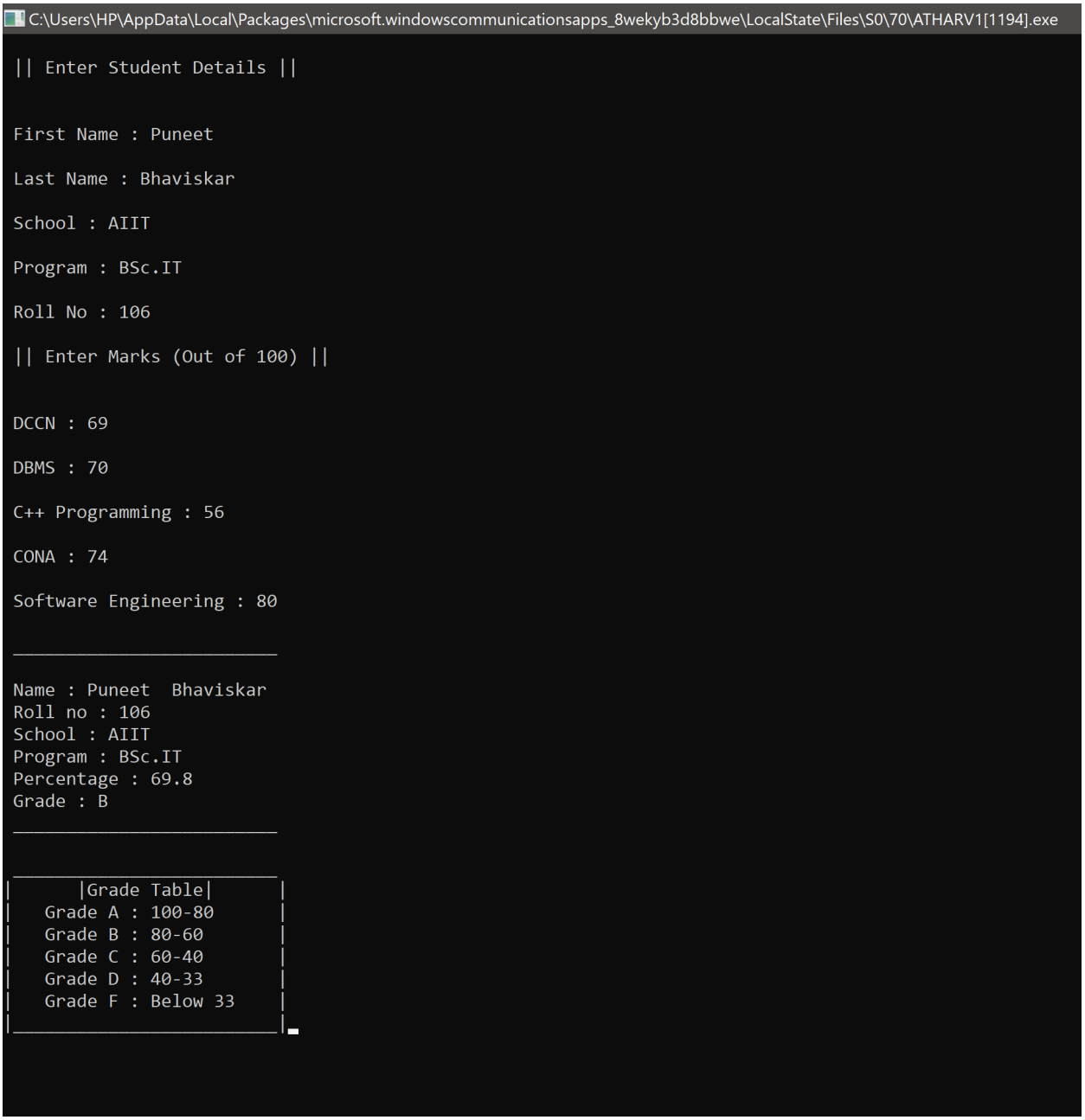


15. Create your own program

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

void getdata();
void displaydata();
void calculate();

class Student
{
    public:
        char F_Name[20], L_Name[20],School[20],Program[20];
        int Roll_no;
        int DCCN, DBMS, CPP, CONA, SE;

            void getdata();
            void displaydata();
            void calculate();
};

void Student::getdata(){

    cout<<"\n || Enter Student Details || \n ";
    cout<<"\n\n First Name : ";
    gets(F_Name);
    cout<<"\n Last Name : ";
    gets(L_Name);
    cout<<"\n School : ";
    gets(School);
    cout<<"\n Program : ";
    gets(Program);
    cout<<"\n Roll No : ";
    cin>>Roll_no;
    cout<<"\n || Enter Marks (Out of 100) || \n ";
    cout<<"\n\n DCCN : ";
    cin>>DCCN;
    cout<<"\n DBMS : ";
    cin>>DBMS;
    cout<<"\n C++ Programming : ";
    cin>>CPP;
    cout<<"\n CONA : ";
    cin>>CONA;
    cout<<"\n Software Engineering : ";
    cin>>SE;

}

void Student::calculate(){

    float Percentage=0;
    Percentage=(DCCN+DBMS+CPP+CONA+SE)/5.0;
    cout<<"\n Percentage : "<<Percentage<<endl;
```

```
54.     if((Percentage>=80)&&(Percentage<=100))
55.     cout<<" Grade : A ";
56.     else if((Percentage>=60)&&(Percentage<80))
57.     cout<<" Grade : B ";
58.     else if((Percentage>=40)&&(Percentage<60))
59.     cout<<" Grade : C ";
60.     else if((Percentage>=33)&&(Percentage<40))
61.     cout<<" Grade : D ";
62.     else
63.     cout<<" Grade : F ";
64.     cout<<"\n _____________________________";
65.     cout<<"\n";
66. }
67. void Student::displaydata(){
68.     cout<<"\n _____________________________";
69.     cout<<"\n\n Name : "<<F_Name<<" "<<L_Name<<" ";
70.     cout<<"\n Roll no : "<<Roll_no;
71.     cout<<"\n School : "<<School;
72.     cout<<"\n Program : "<<Program;
73.
74.
75. }
76.
77. main(){
78.     Student S;
79.     S.getdata();
80.     S.displaydata();
81.     S.calculate();
82.     cout<<"\n _____________________________";
83.     cout<<"\n|       |Grade Table|       |";
84.     cout<<"\n|    Grade A : 100-80       |";
85.     cout<<"\n|    Grade B : 80-60        |";
86.     cout<<"\n|    Grade C : 60-40        |";
87.     cout<<"\n|    Grade D : 40-33        |";
88.     cout<<"\n|    Grade F : Below 33     |";
89.     cout<<"\n|___________________________|";
90.     getch();
91.
92. }
```

Output:


C:\Users\HP\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Files\S0\70\ATHARV1[1194].exe
|| Enter Student Details ||
First Name : Puneet
Last Name : Bhaviskar
School : AIIT
Program : BSc.IT
Roll No : 106
|| Enter Marks (Out of 100) ||
DCCN : 69
DBMS : 70
C++ Programming : 56
CONA : 74
Software Engineering : 80
Name : Puneet Bhaviskar
Roll no : 106
School : AIIT
Program : BSc.IT
Percentage : 69.8
Grade : B
|Grade Table|
Grade A : 100-80
Grade B : 80-60
Grade C : 60-40
Grade D : 40-33
Grade F : Below 33