

1 . Define a Class String. Write overload function == compare two strings

The screenshot shows a C++ IDE with a project named "equal or not using overload.cpp". The code defines a class `myClass` with a private member `int value;` and a public member function `myClass(int val) : value(val) {}`. It also overloads the `==` operator as a const member function: `bool operator==(const myClass& other) const { return this->value == other.value; }`. The `main` function creates two objects, `obj1` and `obj2`, with values 3 and 5 respectively, and compares them using `obj1 == obj2`. The output window shows the program's execution: "enter the first:3", "enter the second:5", "not equal:", "Process exited after 3.491 seconds with return value 0", and "Press any key to continue . . .". The compilation results show 0 errors and 0 warnings.

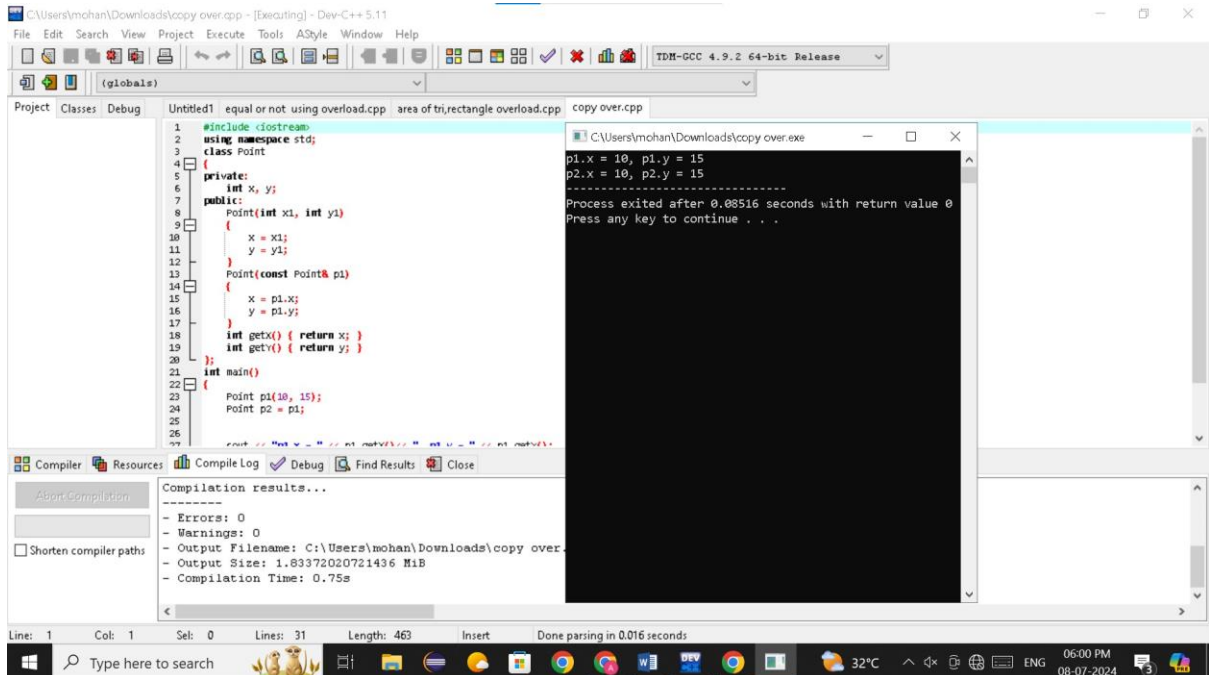
```
1 #include <iostream>
2 class myClass
3 {
4 private:
5     int value;
6 public:
7     myClass(int val) : value(val) {}
8     bool operator==(const myClass& other) const
9     {
10         return this->value == other.value;
11     }
12 };
13
14 int main() {
15     myClass obj1(3);
16     myClass obj2(5);
17     if (obj1 == obj2) {
18         std::cout << "obj1 is equal to obj2\n";
19     } else {
20         std::cout << "obj1 is not equal to obj2\n";
21     }
22     if (obj1 == obj2)
23     }
```

2. Write a program to find area of circle, rectangle and triangle using constructor overloading.

The screenshot shows a C++ IDE with a project named "area of tri,rectangle overload.cpp". The code defines a class `Shape` with a private member `double area;` and a public member function `Shape(double radius)`. It also overloads the `==` operator as a const member function: `bool operator==(const Shape& other) const { return this->area == other.area; }`. The `main` function creates three objects, `obj1`, `obj2`, and `obj3`, with areas 12.5664, 6, and 20 respectively, and compares them using `obj1 == obj2` and `obj2 == obj3`. The output window shows the program's execution: "Enter radius of circle: 2", "Enter length and breadth of rectangle: 2", "Enter base and height of triangle: 4", "Area of Circle is: 12.5664", "Area of Rectangle is: 6", "Area of Rectangle is: 20", "Process exited after 4.111 seconds with return value 0", and "Press any key to continue . . .". The compilation results show 0 errors and 0 warnings.

```
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4 class Shape
5 {
6 public:
7     Shape() {}
8     Shape(double radius)
9     {
10         area = M_PI * radius * radius;
11         shapeType = "Circle";
12     }
13     Shape(double length, double breadth, double base, double height)
14     {
15         if (isRectangle) {
16             area = length * breadth;
17             shapeType = "rectangle";
18         }
19         else {
20             area = 0;
21             shapeType = "Unknown";
22         }
23     }
24     Shape(double base, double height, int type)
25     {
26         if (isTriangle) {
27             area = 0.5 * base * height;
28             shapeType = "triangle";
29         }
30     }
31     bool operator==(const Shape& other) const
32     {
33         return this->area == other.area;
34     }
35 }
```

3. Write a C++ program to demonstrate the working of a copy constructor.



```
#include <iostream>
using namespace std;
class Point
{
private:
    int x, y;
public:
    Point(int x1, int y1)
    {
        x = x1;
        y = y1;
    }
    Point(const Point& p1)
    {
        x = p1.x;
        y = p1.y;
    }
    int getx() { return x; }
    int gety() { return y; }
};

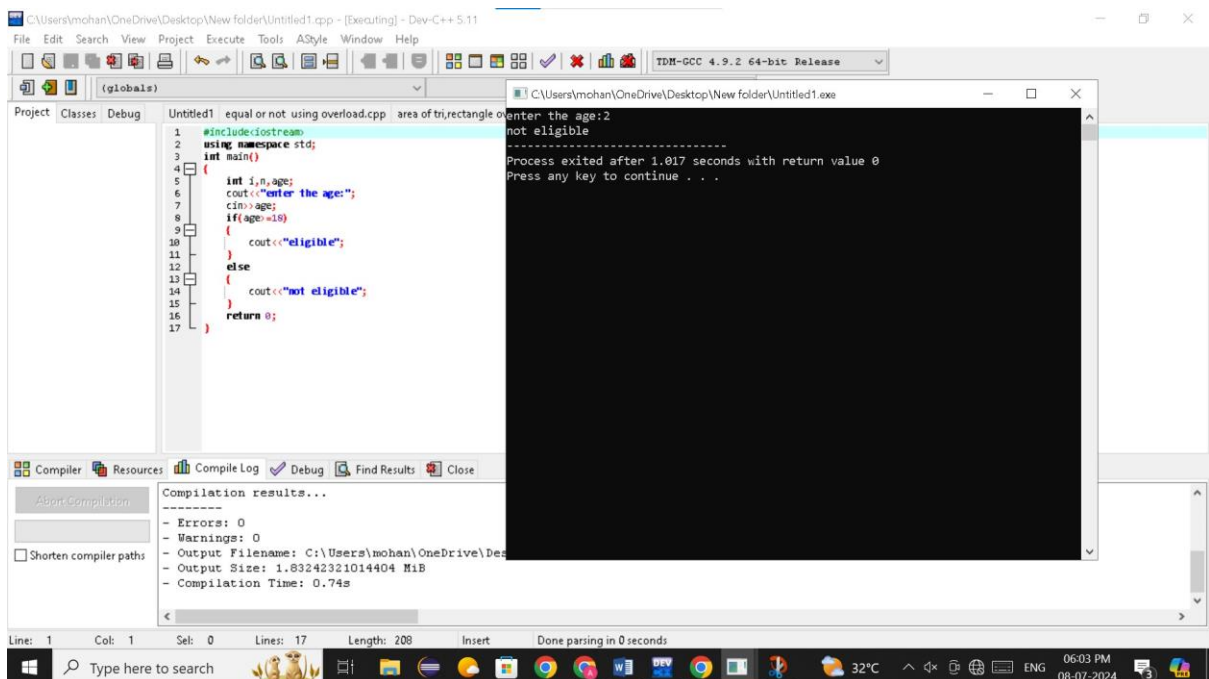
int main()
{
    Point p1(10, 15);
    Point p2 = p1;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\Downloads\copy over.exe
- Output Size: 1.83372020721436 KiB
- Compilation Time: 0.75s

Process exited after 0.08516 seconds with return value 0  
Press any key to continue . . .

4. Write a C++ program to find whether the person is eligible for vote or not.



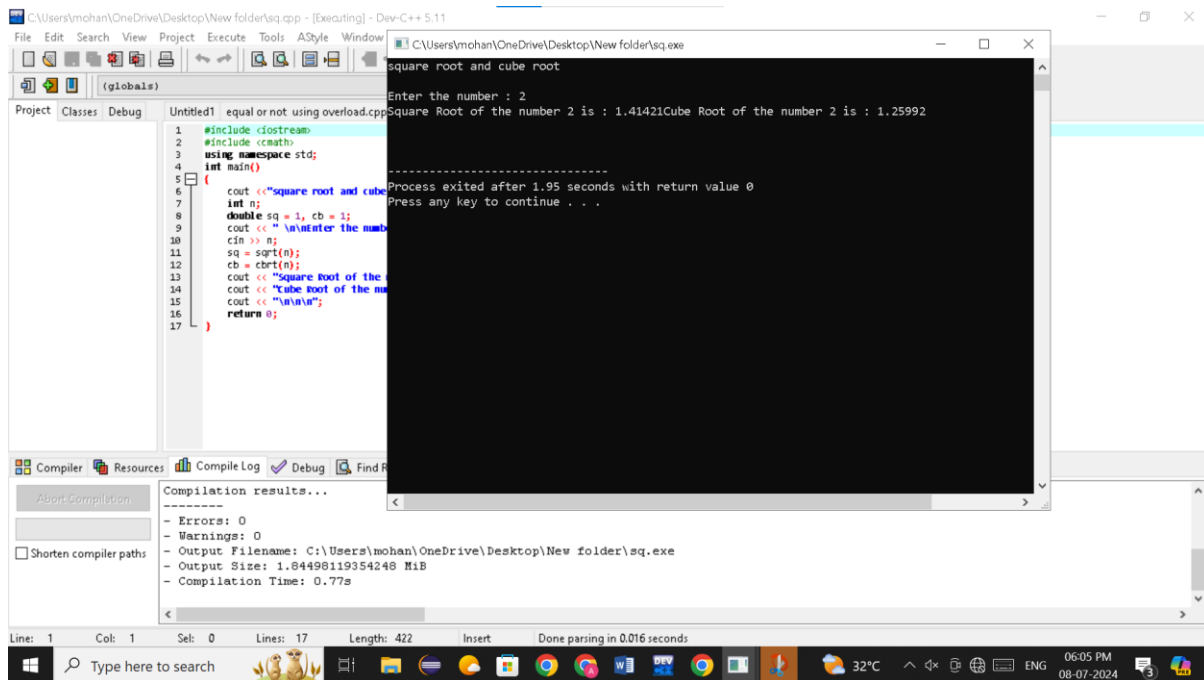
```
#include <iostream>
using namespace std;
int main()
{
    int i, n, age;
    cout << "enter the ages:";
    cin >> age;
    if (age >= 18)
    {
        cout << "eligible";
    }
    else
    {
        cout << "not eligible";
    }
    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\OneDrive\Desktop\New folder\Untitled1.exe
- Output Size: 1.83242321014404 KiB
- Compilation Time: 0.74s

enter the age:2  
not eligible  
Process exited after 1.017 seconds with return value 0  
Press any key to continue . . .

5. Write a CPP program to find the Square root and Cube root of a number.



The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter a number, calculates its square root and cube root, and displays the results. The output window shows the execution results for the input number 2.

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    cout << "square root and cube root\n";
    int n;
    double sq = 1, cb = 1;
    cout << "Enter the number\n";
    cin >> n;
    sq = sqrt(n);
    cb = cbrt(n);
    cout << "Square root of the number is : " << sq << "\n";
    cout << "Cube root of the number is : " << cb << "\n";
    return 0;
}
```

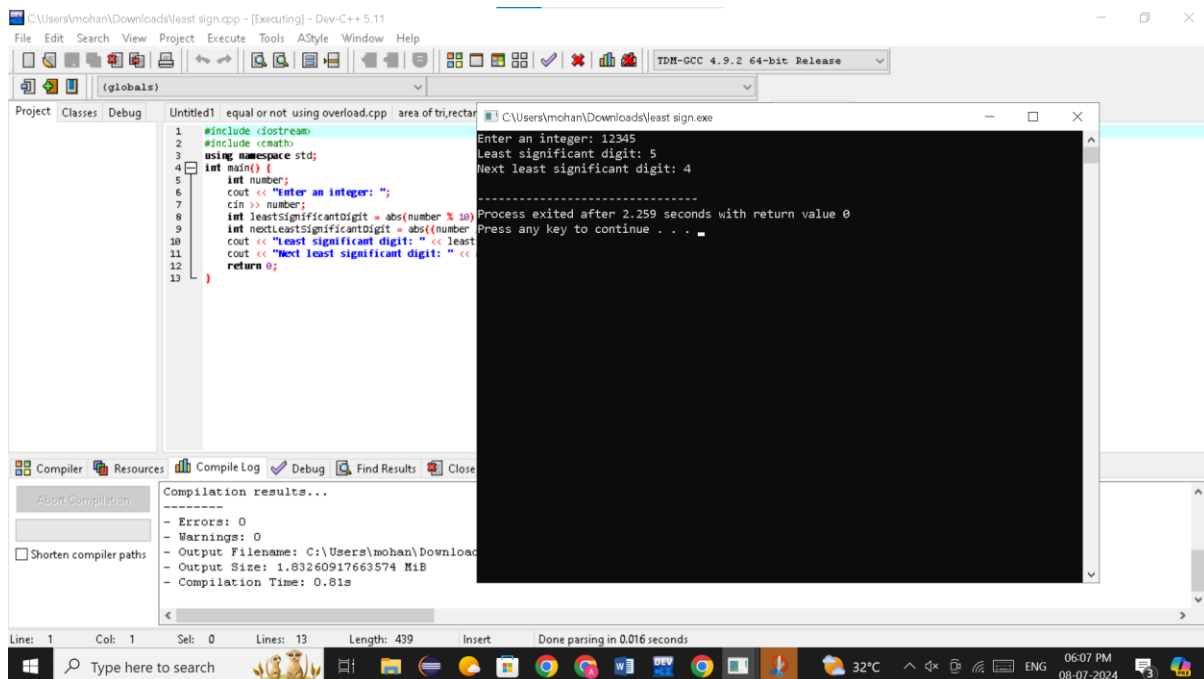
Output:

```
Enter the number : 2
Square Root of the number 2 is : 1.41421
Cube Root of the number 2 is : 1.25992
Process exited after 1.95 seconds with return value 0
Press any key to continue . . .
```

Compilation results:

```
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\OneDrive\Desktop\New folder\sqr.exe
- Output Size: 1.84498119354248 MiB
- Compilation Time: 0.77s
```

6. Write a C++ program that reads an integer and prints the least significant digit and the next least significant digit.



The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter an integer, calculates the least significant digit and the next least significant digit, and displays the results. The output window shows the execution results for the input integer 12345.

```
#include <iostream>
#include <cmath>
using namespace std;
int main()
{
    cout << "Enter an integer: ";
    int number;
    cin >> number;
    int leastSignificantDigit = abs(number % 10);
    int nextLeastSignificantDigit = abs((number / 10) % 10);
    cout << "Least significant digit: " << leastSignificantDigit << "\n";
    cout << "Next least significant digit: " << nextLeastSignificantDigit << "\n";
    return 0;
}
```

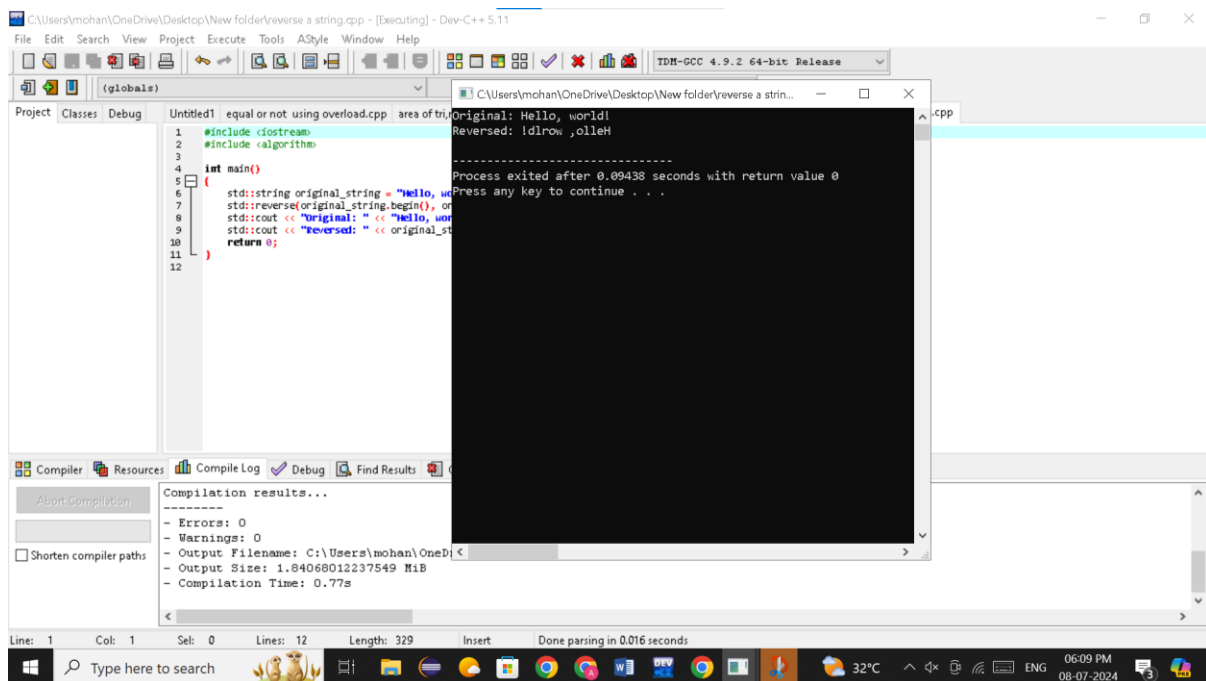
Output:

```
Enter an integer: 12345
Least significant digit: 5
Next least significant digit: 4
Process exited after 2.259 seconds with return value 0
Press any key to continue . . .
```

Compilation results:

```
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\Downloads\least sign.exe
- Output Size: 1.83260917663574 MiB
- Compilation Time: 0.81s
```

7. Write a program in C++ to print a string in reverse using a pointer.



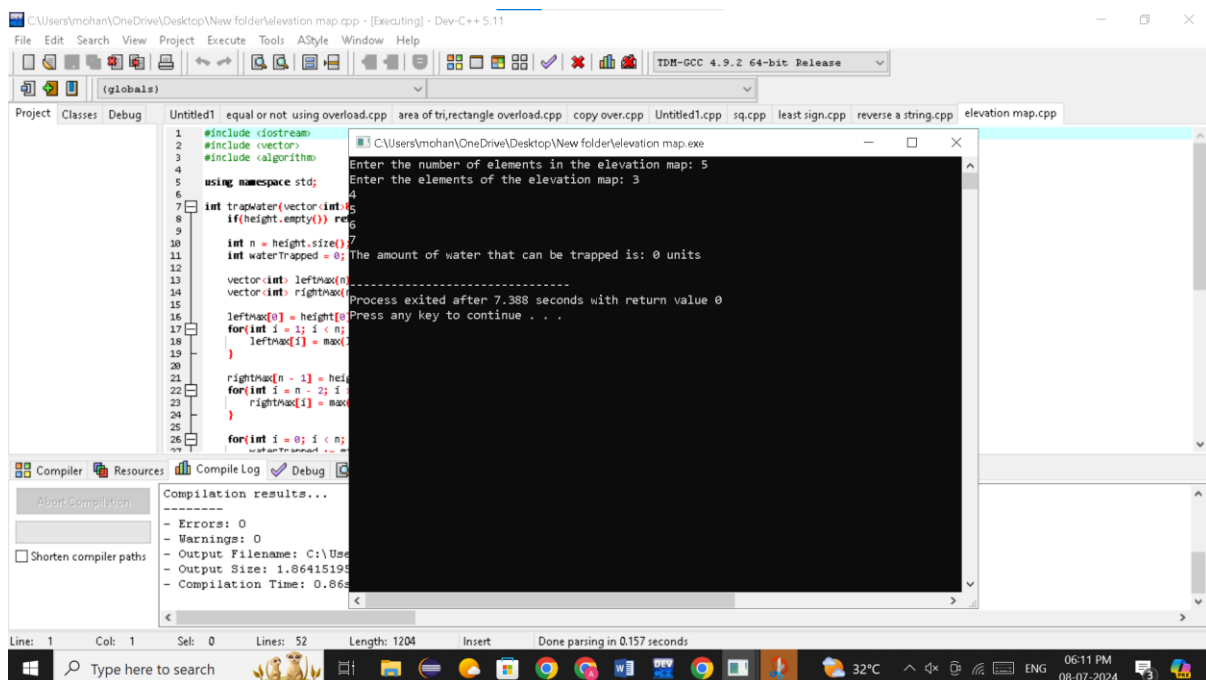
```
#include <iostream>
#include <algorithm>

int main()
{
    std::string original_string = "Hello, world!";
    std::reverse(original_string.begin(), original_string.end());
    std::cout << "Original: " << original_string << "\n";
    std::cout << "Reversed: " << original_string << "\n";
    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\OneDrive\Desktop\New folder\reverse a string.exe
- Output Size: 1.84068012237549 MiB
- Compilation Time: 0.77s

8. Write a C++ Program to check whether a given number is palindrome or not. Given an integer array A[ ] consisting of N non-negative integers representing an elevation map:



```
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

int trapWater(vector<int> height)
{
    if(height.empty()) return 0;
    int n = height.size();
    int waterTrapped = 0;
    vector<int> leftMax(n);
    vector<int> rightMax(n);
    leftMax[0] = height[0];
    for(int i = 1; i < n; i++)
        leftMax[i] = max(leftMax[i-1], height[i]);
    rightMax[n-1] = height[n-1];
    for(int i = n-2; i >= 0; i--)
        rightMax[i] = max(rightMax[i+1], height[i+1]);
    for(int i = 0; i < n; i++)
        waterTrapped += min(leftMax[i], rightMax[i]) - height[i];
    return waterTrapped;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\OneDrive\Desktop\New folder\elevation map.exe
- Output Size: 1.86415195 MiB
- Compilation Time: 0.86s

9. write a C++ Program to check whether a given number is palindrome or not.

The screenshot shows a C++ program in Dev-C++ that checks if a number is a palindrome. The program is named 'palindrome (num.cpp)' and is located at 'C:\Users\mohan\Downloads\palindrome (num.cpp)'. The code is as follows:

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int n, rev=0, remainder, original;
6     cout<<"Enter an integer: ";
7     cin>>n;
8     original = n;
9     while (n != 0)
10    {
11        remainder = n % 10;
12        rev= rev*10 + remainder;
13        n /= 10;
14    }
15    if (original == rev)
16        cout<<"palindrome.";
17    else
18        cout<<"not a palindrome.";
19
20    return 0;
21 }
```

The program is executed, and the output is shown in the console window:

```
Enter an integer: 123
not a palindrome.
Process exited after 1.959 seconds with return value 0
Press any key to continue . . .
```

The compilation results are also shown in the 'Compiler' window:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\mohan\Downloads\palindrome (num.exe)
- Output Size: 1.83244228363037 MiB
- Compilation Time: 0.73s
```