**Assignment-3**

1. Write a C++ program to demonstrate the overloading of a unary operator.

#include <iostream>

class MyNumber {

private:

int value;

public:

MyNumber(int val) : value(val) {}

MyNumber operator-() {

return MyNumber(-value);

}

void display() {

std::cout << "Value: " << value << std::endl;

}

};

int main() {

MyNumber num1(5);

MyNumber num2 = -num1;

std::cout << "Original Number:" << std::endl;

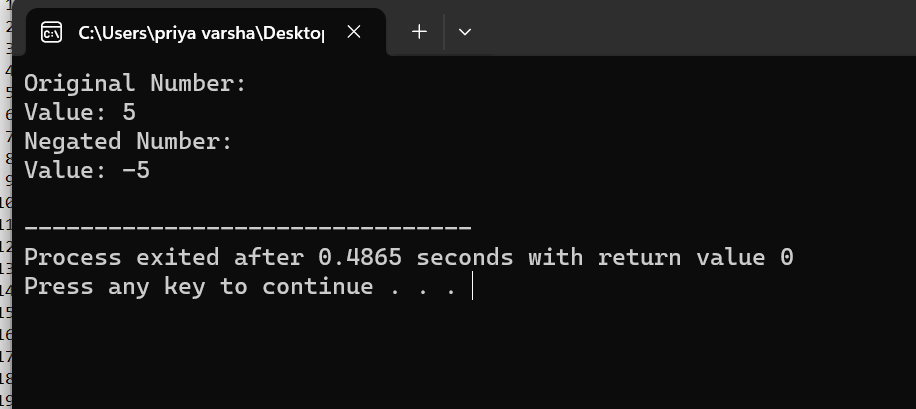
num1.display();

std::cout << "Negated Number:" << std::endl;

num2.display();

return 0;

}



2) Write a C++ program to demonstrate the overloading of a binary operator.

#include <iostream>

class ComplexNumber {

private:

double real;

double imag;

public:

ComplexNumber(double r, double i) : real(r), imag(i) {}

ComplexNumber operator+(const ComplexNumber& other) {

double newReal = real + other.real;

double newImag = imag + other.imag;

return ComplexNumber(newReal, newImag);

}

void display() {

std::cout << real << " + " << imag << "i" << std::endl;

}

};

int main() {

ComplexNumber num1(2.0, 3.0);

ComplexNumber num2(1.5, 2.5);

ComplexNumber sum = num1 + num2;

std::cout << "Num1: ";

num1.display();

std::cout << "Num2: ";

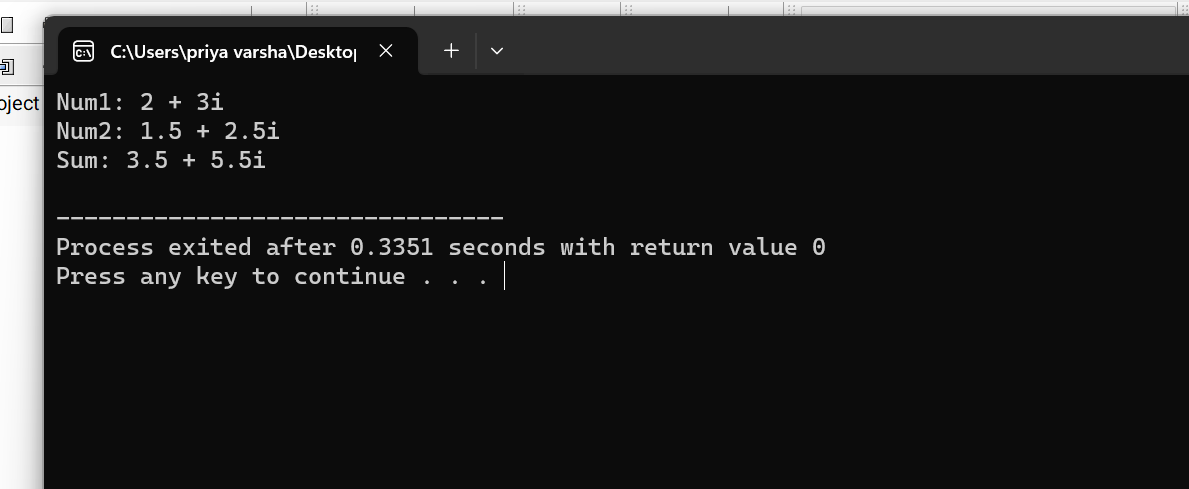
num2.display();

std::cout << "Sum: ";

sum.display();

return 0;

}



3)What is the output of the following code?

#include <iostream>

#include <cstring>

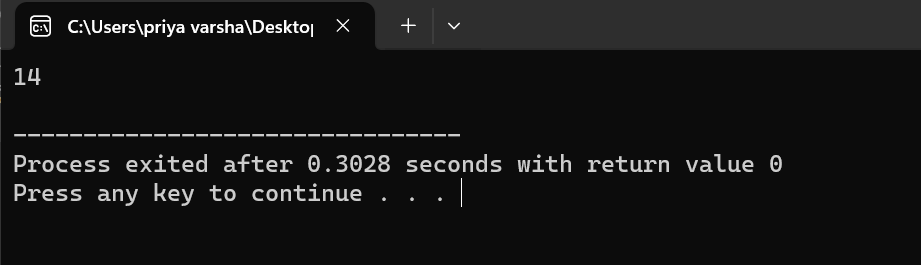
int main() {

const char\* myString = "Hello, World.\n";

std::cout << std::strlen(myString) << "\n";

return 0;

}



1. What is the output of the following code?

#include <iostream>

int main() {

int a = 20;

int &n = a;

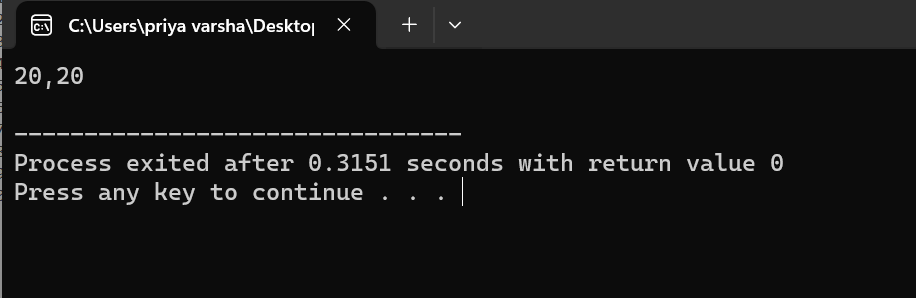
n = a++;

a = n++;

std::cout << a << "," << n << std::endl;

return 0;

}



1. Write down a C++ program to implement function overloading.

#include <iostream>

int add(int a, int b) {

return a + b;

}

double add(double a, double b) {

return a + b;

}

std::string add(std::string str1, std::string str2) {

return str1 + str2;

}

int main() {

int intResult = add(5, 3);

double doubleResult = add(3.5, 2.7);

std::string stringResult = add("Hello, ", "World!");

std::cout << "Sum of integers: " << intResult << std::endl;

std::cout << "Sum of doubles: " << doubleResult << std::endl;

std::cout << "Concatenated string: " << stringResult << std::endl;

return 0;

}

