1.Write a program to read in two integers and perform the following operations on them: addition, subtraction, multiplication, division, and modulo.

Program:

#include <iostream>

using namespace std;

int main() {

// Define variables

int num1, num2, sum, difference, product, quotient;

// Input two numbers

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Perform arithmetic operations

sum = num1 + num2;

difference = num1 - num2;

product = num1 \* num2;

// Check if num2 is not zero to avoid division by zero

if (num2 != 0) {

quotient = num1 / num2;

cout << "Quotient: " << quotient << endl;

} else {

cout << "Cannot divide by zero!" << endl;

}

// Output results

cout << "Sum: " << sum << endl;

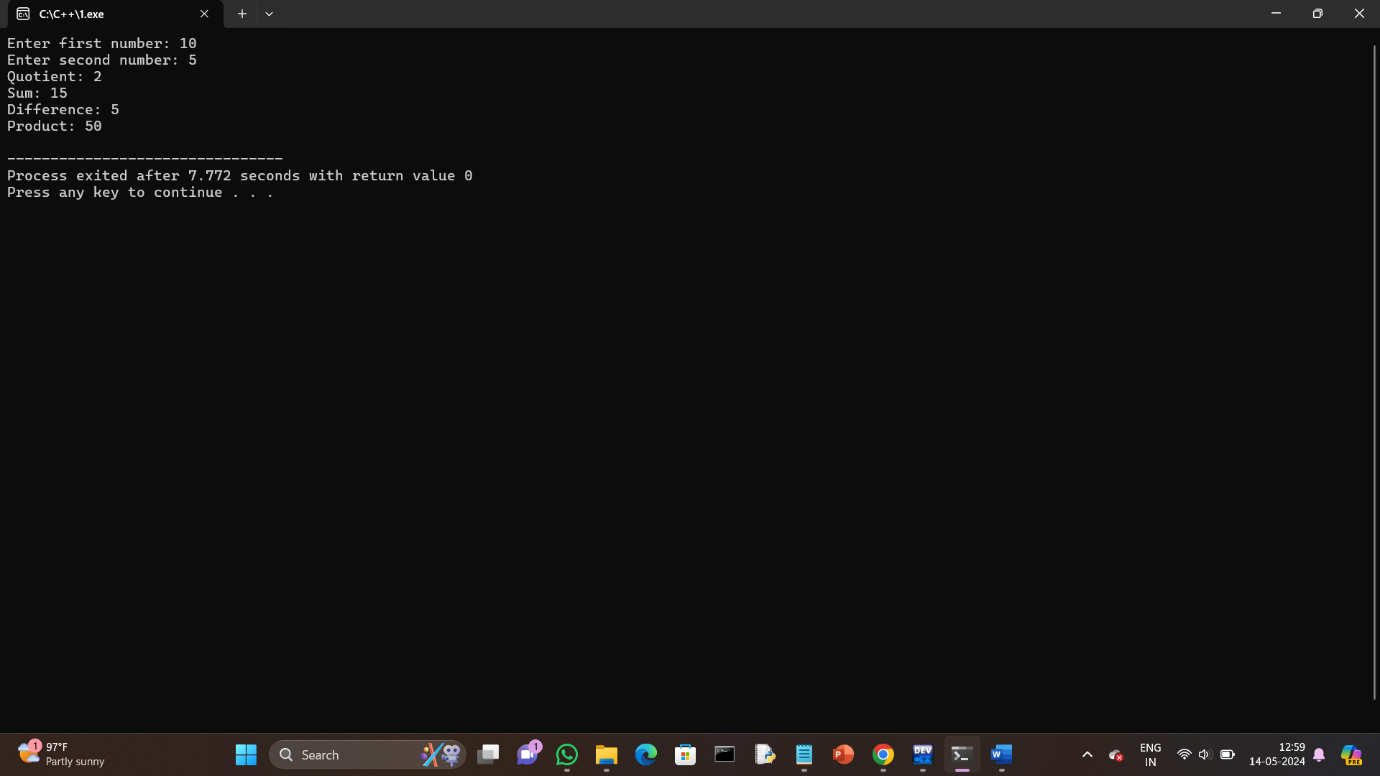
cout << "Difference: " << difference << endl;

cout << "Product: " << product << endl;

return 0;

}

Output:



2.Program to determine the integer is odd or even.

Program:

#include <iostream>

using namespace std;

int main() {

int num;

cout << "Enter an integer: ";

cin >> num;

if(num % 2 == 0) {

cout << num << " is even." << endl;

} else {

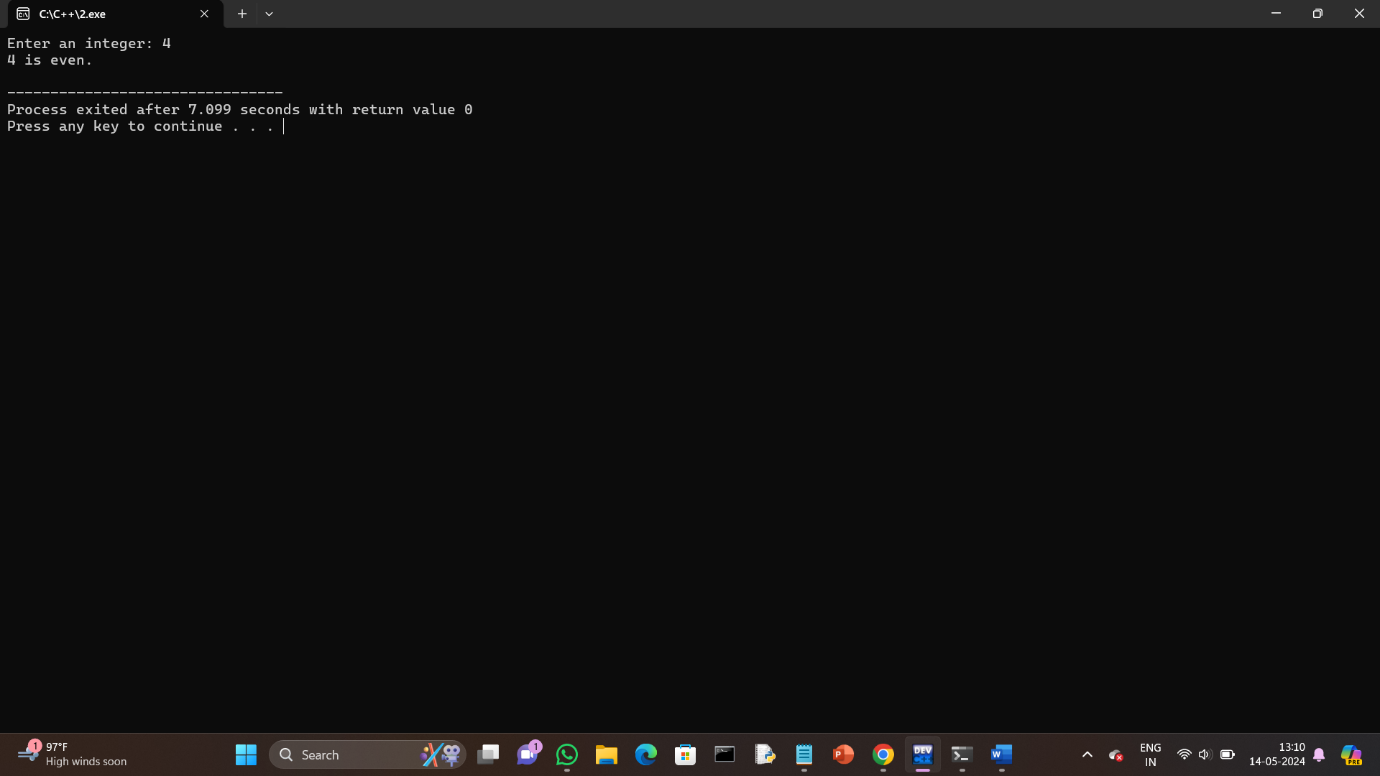
cout << num << " is odd." << endl;

}

return 0;

}

Output:



3.Program to compute the average of three integers.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2, num3;

float average;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

cout << "Enter third number: ";

cin >> num3;

// Compute the average of the three numbers

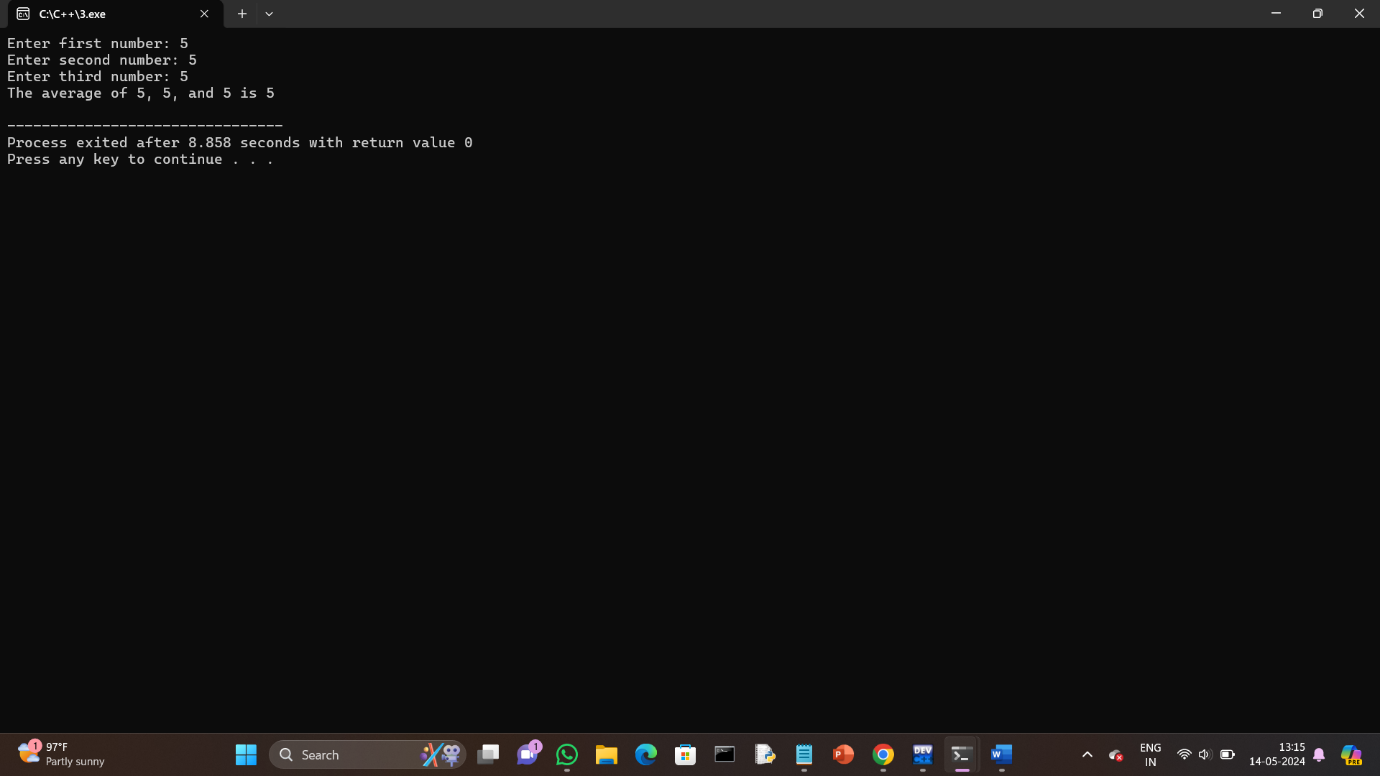
average = (num1 + num2 + num3) / 3.0;

cout << "The average of " << num1 << ", " << num2 << ", and " << num3 << " is " << average << endl;

return 0;

}

Output:



4.Program to check two numbers are equal or not.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Check if the two numbers are equal

if (num1 == num2) {

cout << "The numbers " << num1 << " and " << num2 << " are equal." << endl;

} else {

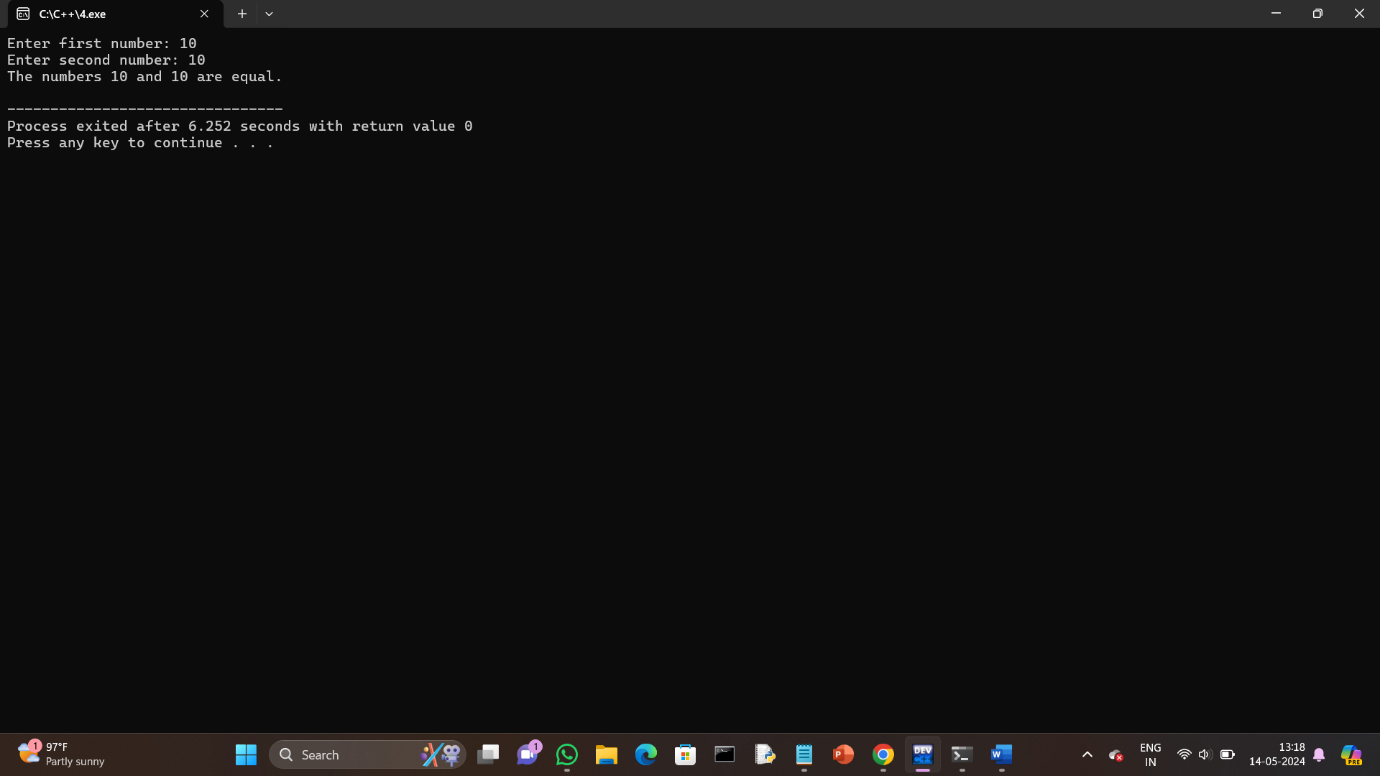
cout << "The numbers " << num1 << " and " << num2 << " are not equal." << endl;

}

return 0;

}

Output:



5.Write a program to read in two Floating numbers and perform the following operations on them: addition, subtraction, multiplication, division, and modulo.

Program:

#include <iostream>

#include <cmath> // For fmod function

using namespace std;

int main() {

float num1, num2;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Perform addition

float addition = num1 + num2;

cout << "Addition: " << num1 << " + " << num2 << " = " << addition << endl;

// Perform subtraction

float subtraction = num1 - num2;

cout << "Subtraction: " << num1 << " - " << num2 << " = " << subtraction << endl;

// Perform multiplication

float multiplication = num1 \* num2;

cout << "Multiplication: " << num1 << " \* " << num2 << " = " << multiplication << endl;

// Perform division

if (num2 != 0) {

float division = num1 / num2;

cout << "Division: " << num1 << " / " << num2 << " = " << division << endl;

} else {

cout << "Error: Division by zero is not allowed." << endl;

}

// Perform modulo operation

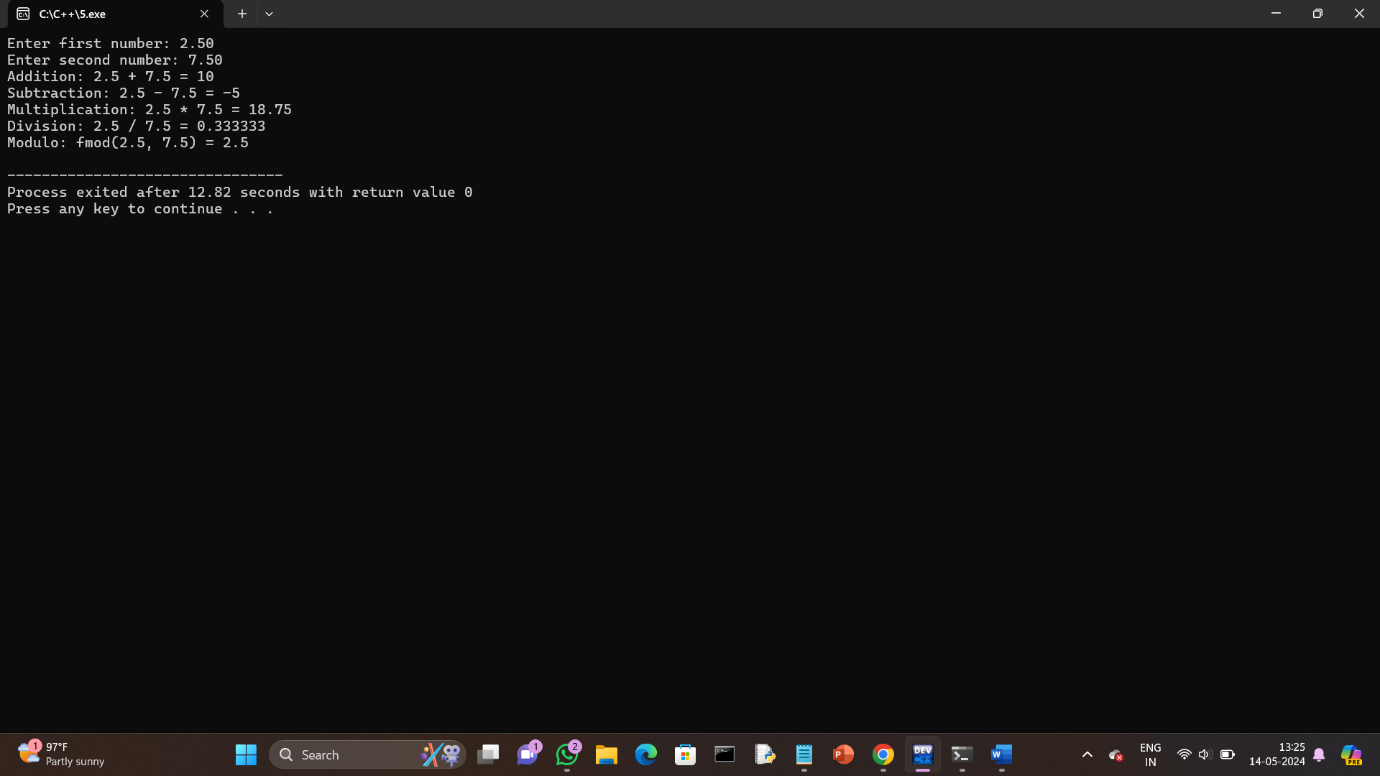
float modulo = fmod(num1, num2);

cout << "Modulo: fmod(" << num1 << ", " << num2 << ") = " << modulo << endl;

return 0;

}

Output:



6.Program to check the character is a vowel or consonant.

Program:

#include <iostream>

using namespace std;

int main() {

char ch;

cout << "Enter a character: ";

cin >> ch;

// Convert character to lowercase to make the check case-insensitive

ch = tolower(ch);

// Check if the character is a vowel

if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {

cout << ch << " is a vowel." << endl;

}

// Check if the character is an alphabet letter

else if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z')) {

cout << ch << " is a consonant." << endl;

}

// If the character is not an alphabet letter

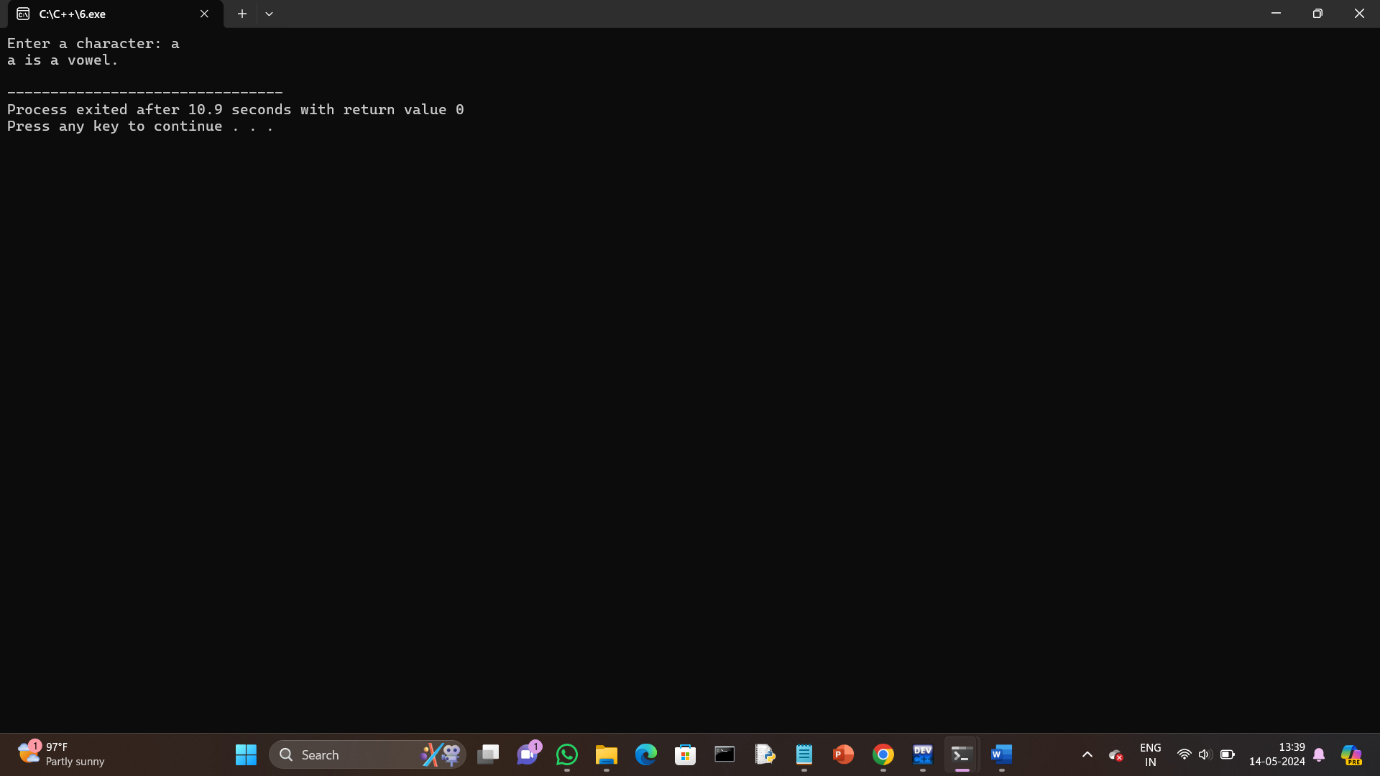
else {

cout << ch << " is not an alphabet letter." << endl;

}

return 0;

Output:



7.Program to check the number is positive, negative or zero.

Program:

#include <iostream>

using namespace std;

int main() {

float num;

cout << "Enter a number: ";

cin >> num;

// Check if the number is positive, negative, or zero

if (num > 0) {

cout << "The number " << num << " is positive." << endl;

} else if (num < 0) {

cout << "The number " << num << " is negative." << endl;

} else {

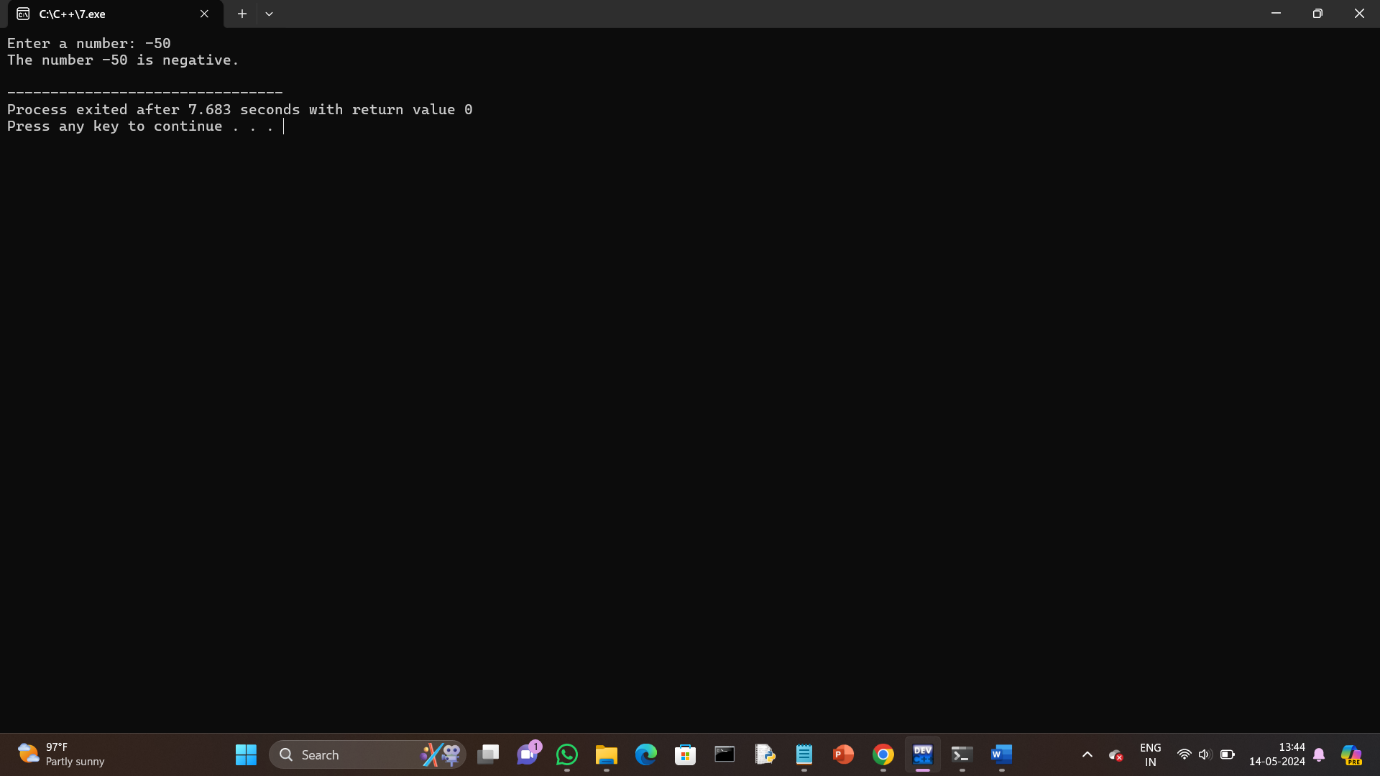
cout << "The number is zero." << endl;

}

return 0;

}

Output:



8.Program to determine which number is greater among two integers.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Determine which number is greater

if (num1 > num2) {

cout << "The greater number is: " << num1 << endl;

} else if (num2 > num1) {

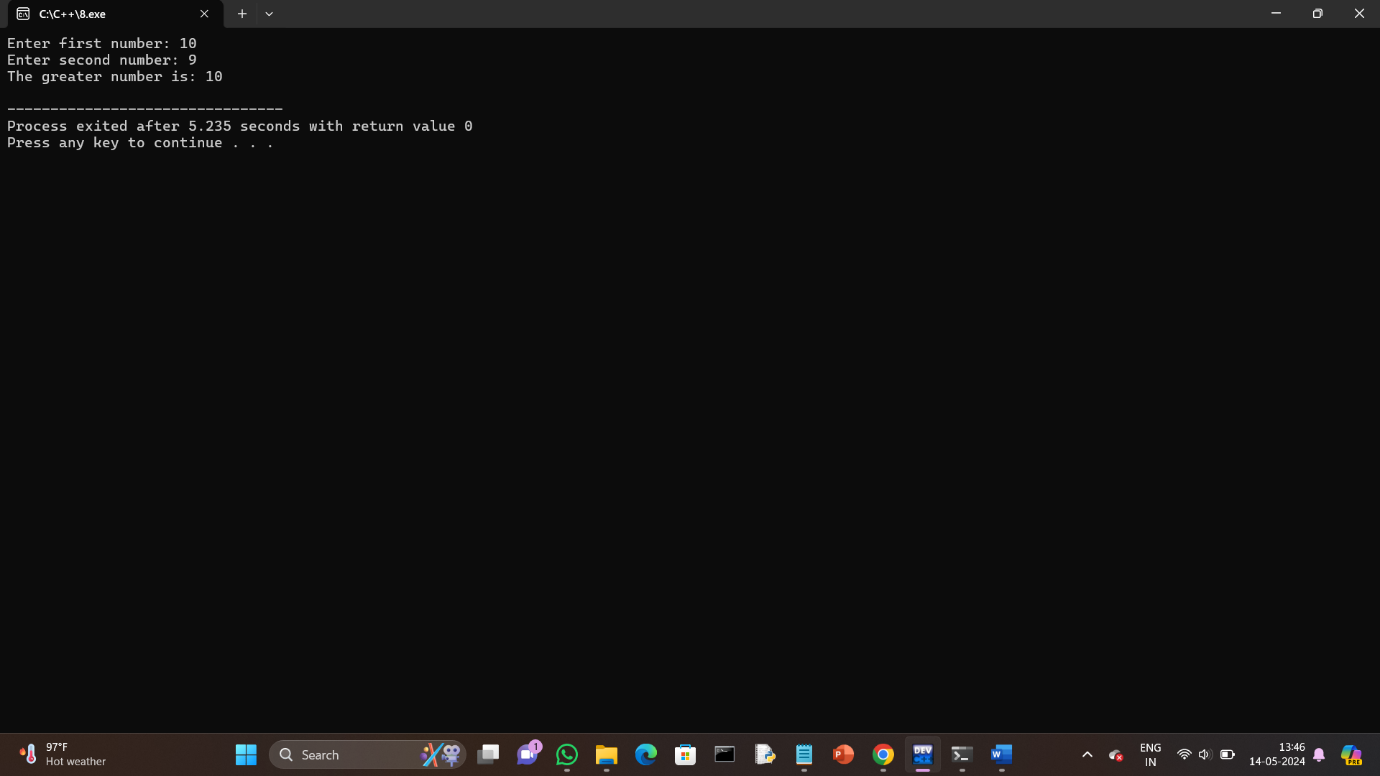
cout << "The greater number is: " << num2 << endl;

} else {

cout << "Both numbers are equal." << endl;

} return 0;

}

Output:

9.Program to read a floating-number and round it to the nearest integer using the floor an ceil functions.

Program:

#include <iostream>

#include <cmath> // For floor and ceil functions

using namespace std;

int main() {

float num;

cout << "Enter a floating-point number: ";

cin >> num;

// Using floor function

int floored = floor(num);

cout << "Using floor, the number rounded down is: " << floored << endl;

// Using ceil function

int ceiled = ceil(num);

cout << "Using ceil, the number rounded up is: " << ceiled << endl;

// Round to the nearest integer using round function

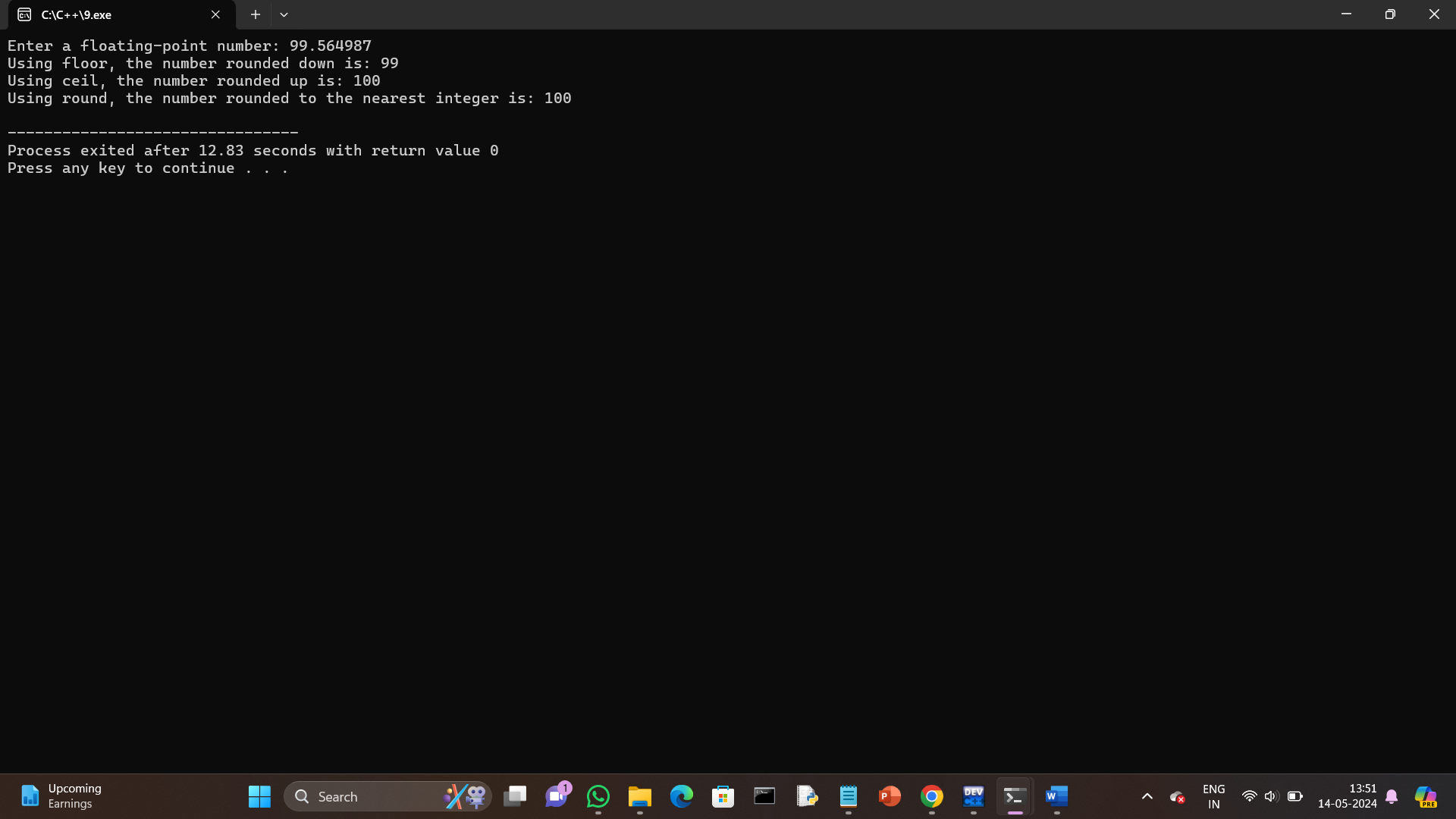
int rounded = round(num);

cout << "Using round, the number rounded to the nearest integer is: " << rounded << endl;

return 0;

}

Output:



10.Program to swap two numbers using bitwise XOR operator.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Display original values

cout << "Before swapping:" << endl;

cout << "num1 = " << num1 << ", num2 = " << num2 << endl;

// Swap numbers using bitwise XOR

num1 = num1 ^ num2;

num2 = num1 ^ num2;

num1 = num1 ^ num2;

// Display swapped values

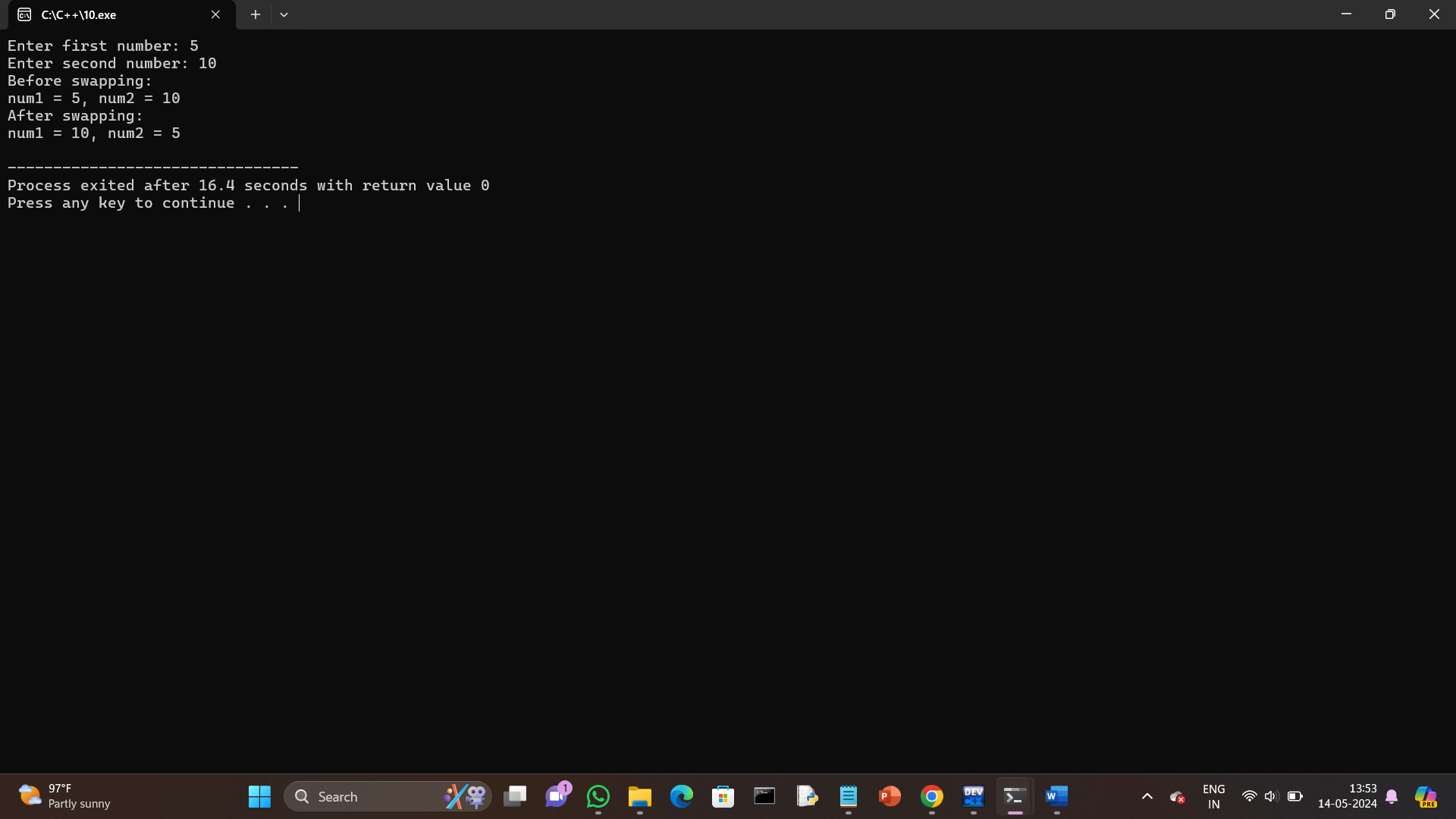
cout << "After swapping:" << endl;

cout << "num1 = " << num1 << ", num2 = " << num2 << endl;

return 0;

}

Output:



11.Largest among three numbers using ternary conditional operator.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2, num3, largest;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

cout << "Enter third number: ";

cin >> num3;

// Using ternary conditional operator to find the largest number

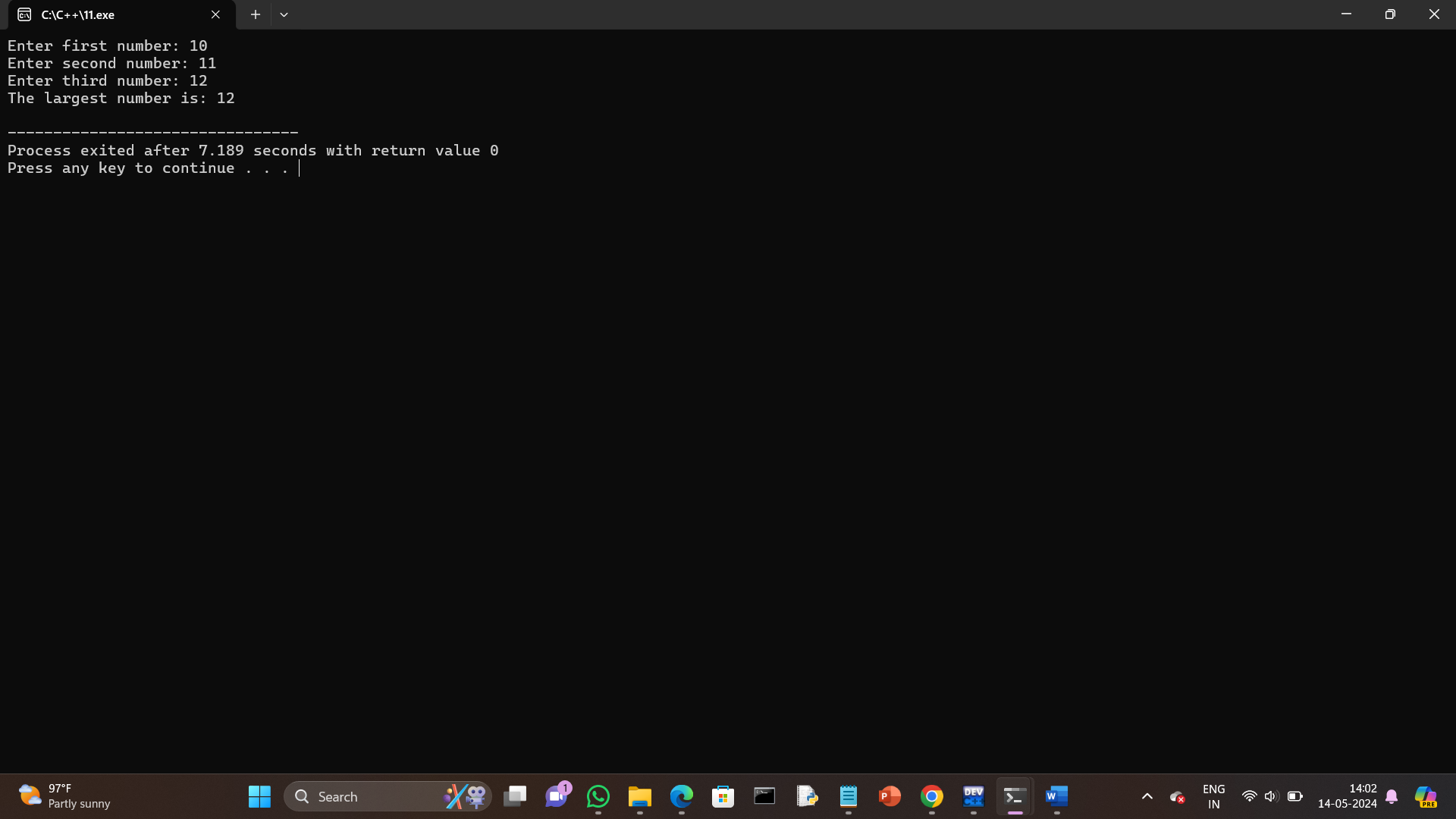
largest = (num1 > num2) ? (num1 > num3 ? num1 : num3) : (num2 > num3 ? num2 : num3);

cout << "The largest number is: " << largest << endl;

return 0;

}

Output:



12.Program to check two numbers are equal or notusing ternary conditional operator.

Program:

#include <iostream>

using namespace std;

int main() {

int num1, num2;

cout << "Enter first number: ";

cin >> num1;

cout << "Enter second number: ";

cin >> num2;

// Check if the two numbers are equal using the ternary conditional operator

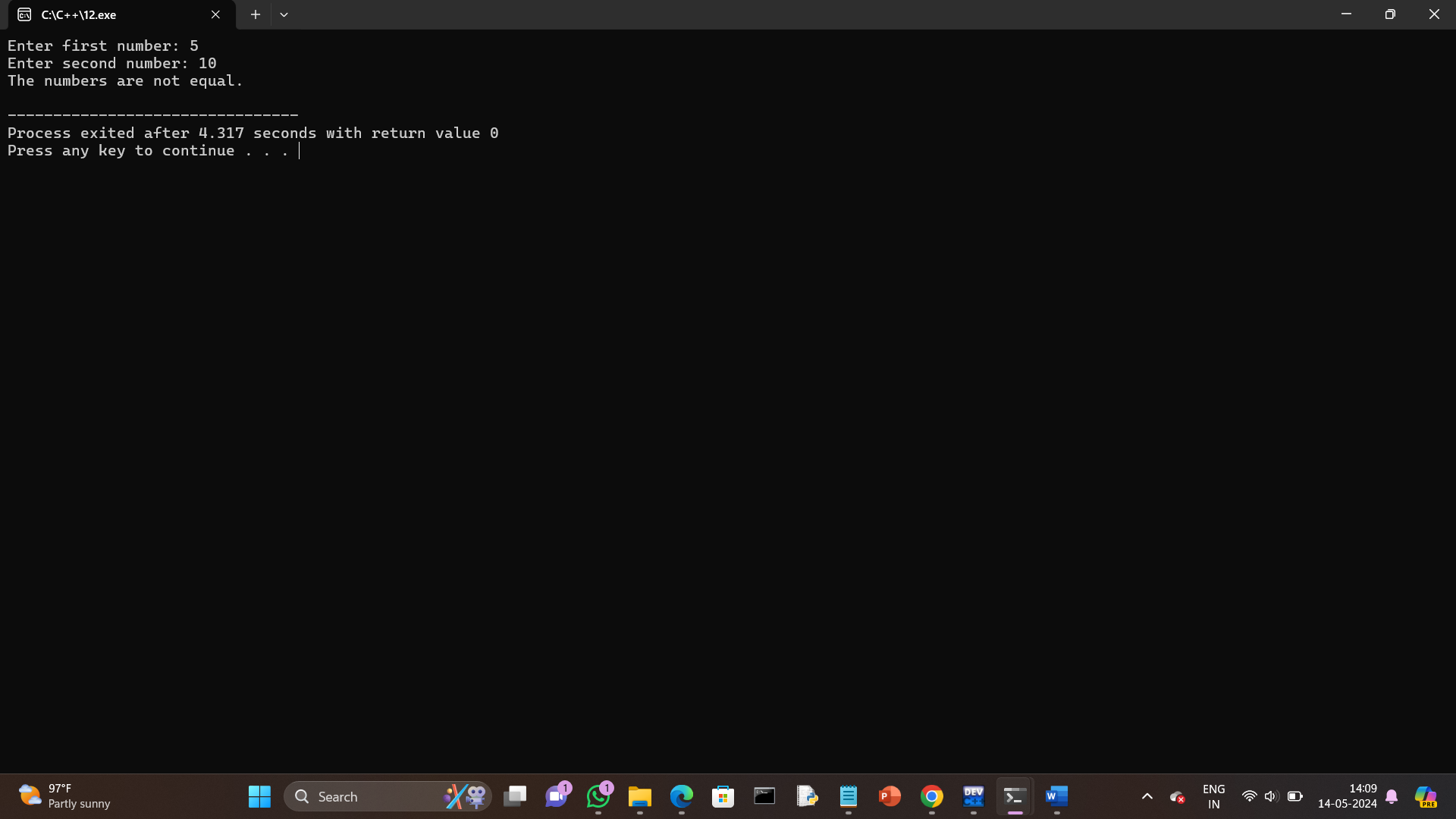
string result = (num1 == num2) ? "The numbers are equal." : "The numbers are not equal.";

cout << result << endl;

return 0;

}

Output:



13.Program to check the integer is divisible by 3 or not using ternary conditional operator.

Program:

#include <iostream>

using namespace std;

int main() {

int num;

cout << "Enter an integer: ";

cin >> num;

// Check if the integer is divisible by 3 using the ternary conditional operator

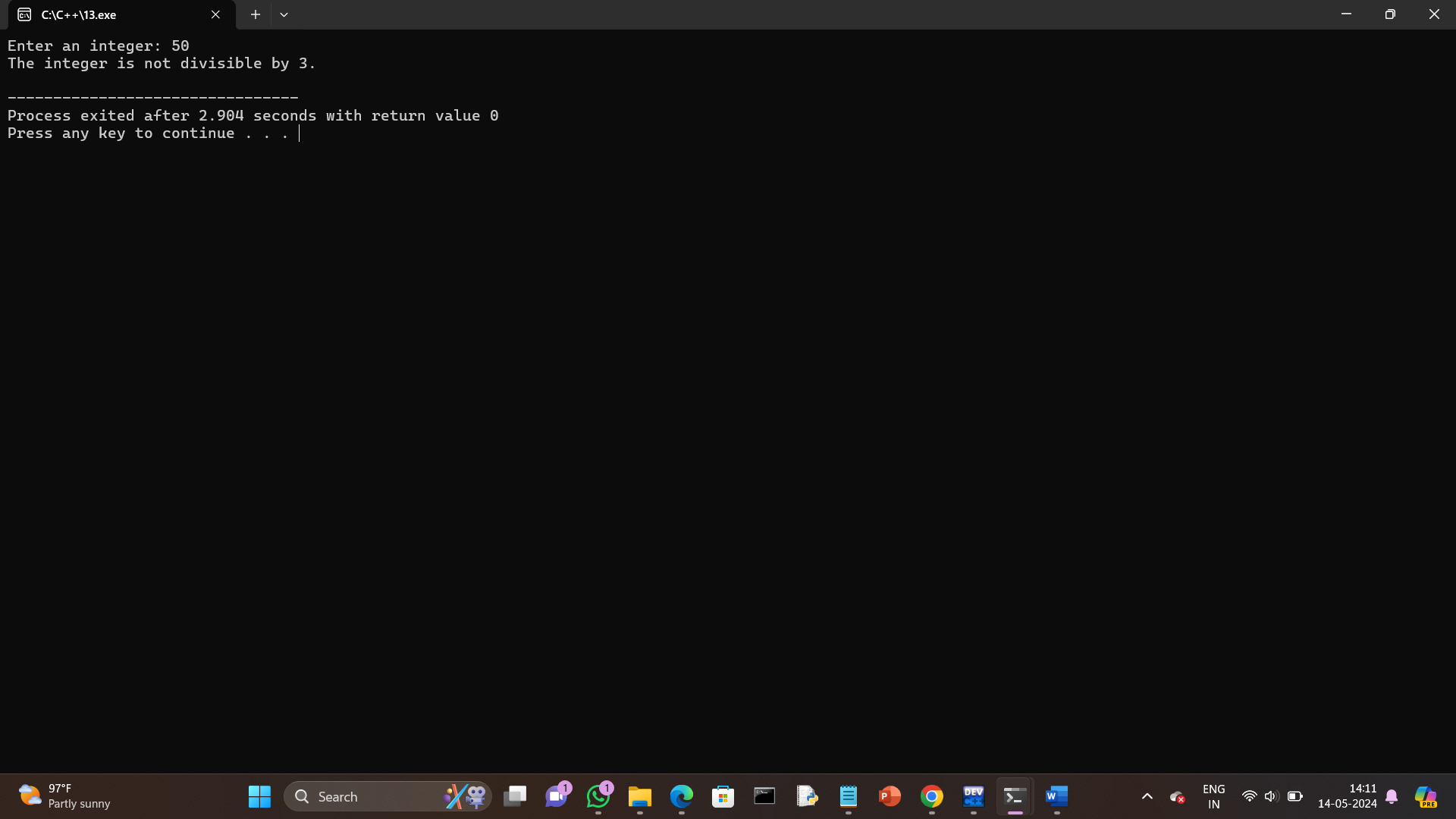
string result = (num % 3 == 0) ? "The integer is divisible by 3." : "The integer is not divisible by 3.";

cout << result << endl;

return 0;

}

Output:



14.Program to print numbers from 1 to 10 using for loop.

Program:

#include <iostream>

using namespace std;

int main() {

// Using a for loop to print numbers from 1 to 10

cout << "Numbers from 1 to 10:" << endl;

for (int i = 1; i <= 10; ++i) {

cout << i << " ";

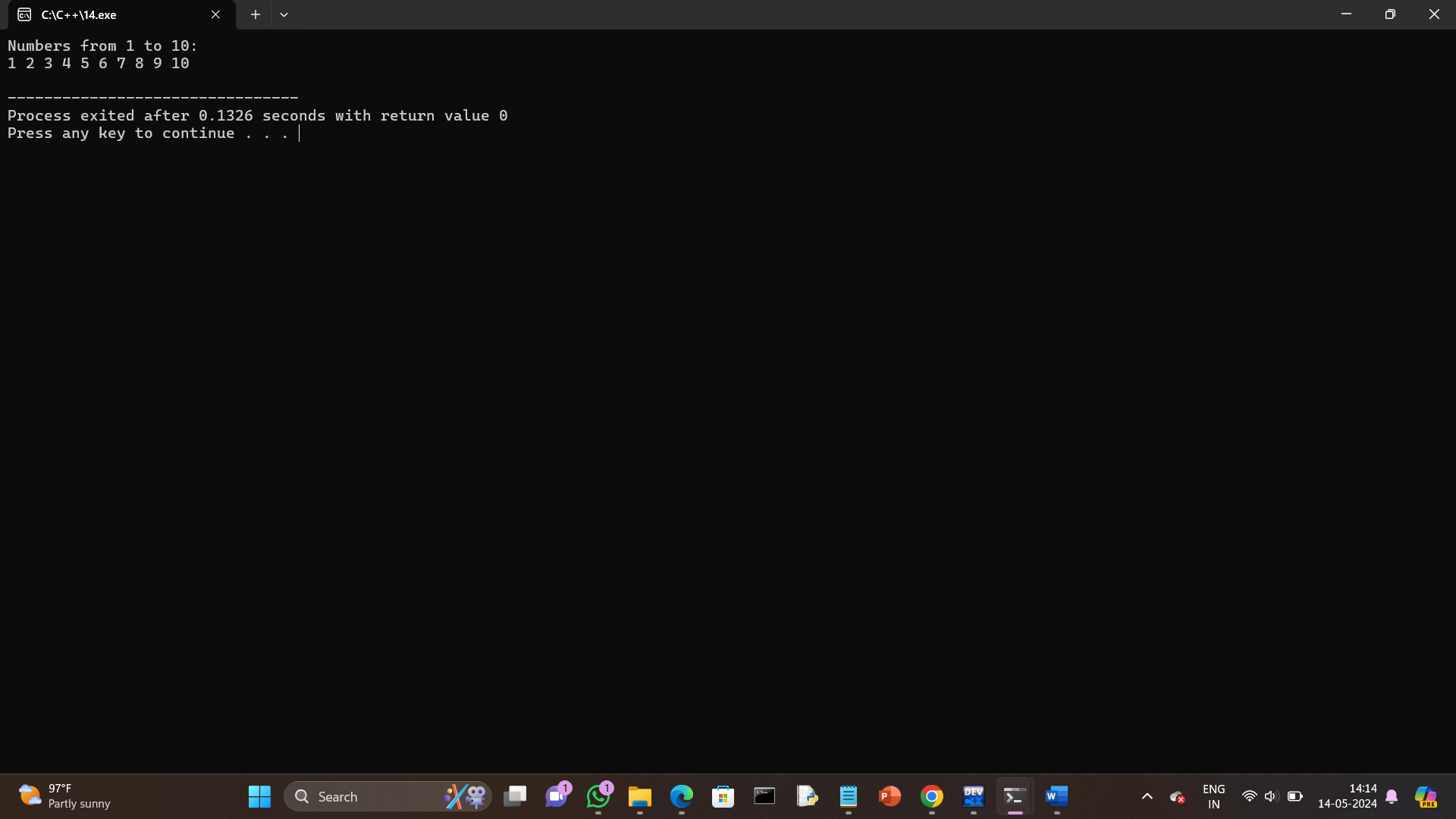
}

cout << endl;

return 0;

}

Output:



15.Factorial of a number using for loop.

Program:

#include <iostream>

using namespace std;

int main() {

int num;

unsigned long long factorial = 1;

cout << "Enter a positive integer: ";

cin >> num;

// Check if the number is negative

if (num < 0) {

cout << "Error: Factorial is not defined for negative numbers." << endl;

return 1;

}

// Calculate factorial using a for loop

for (int i = 1; i <= num; ++i) {

factorial \*= i;

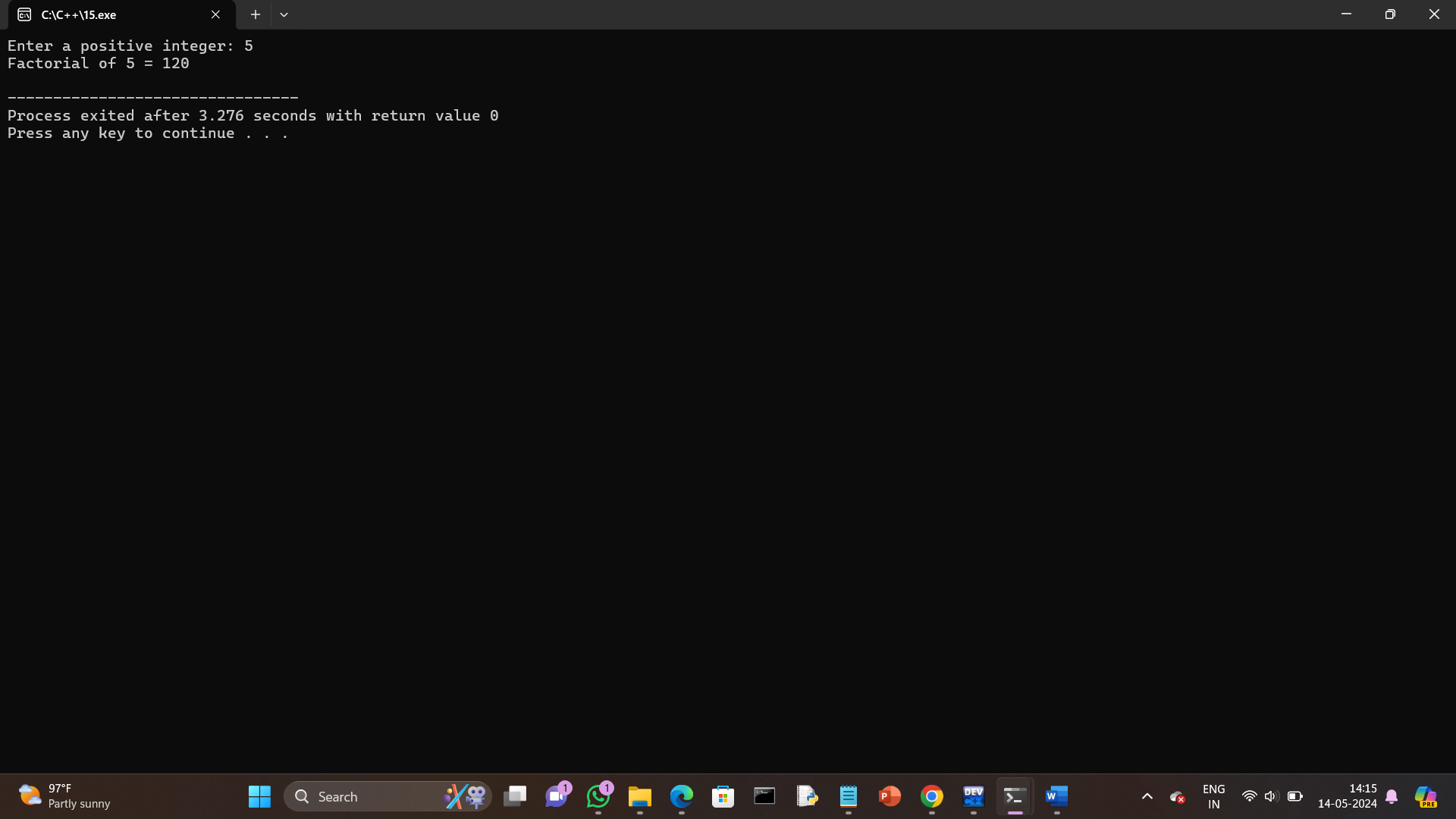
}

cout << "Factorial of " << num << " = " << factorial << endl;

return 0;

}

Output:



16.Print multiplication table using for loop.

Program:

#include <iostream>

using namespace std;

int main() {

int num;

cout << "Enter the number to print its multiplication table: ";

cin >> num;

cout << "Multiplication table of " << num << ":" << endl;

for (int i = 1; i <= 10; ++i) {

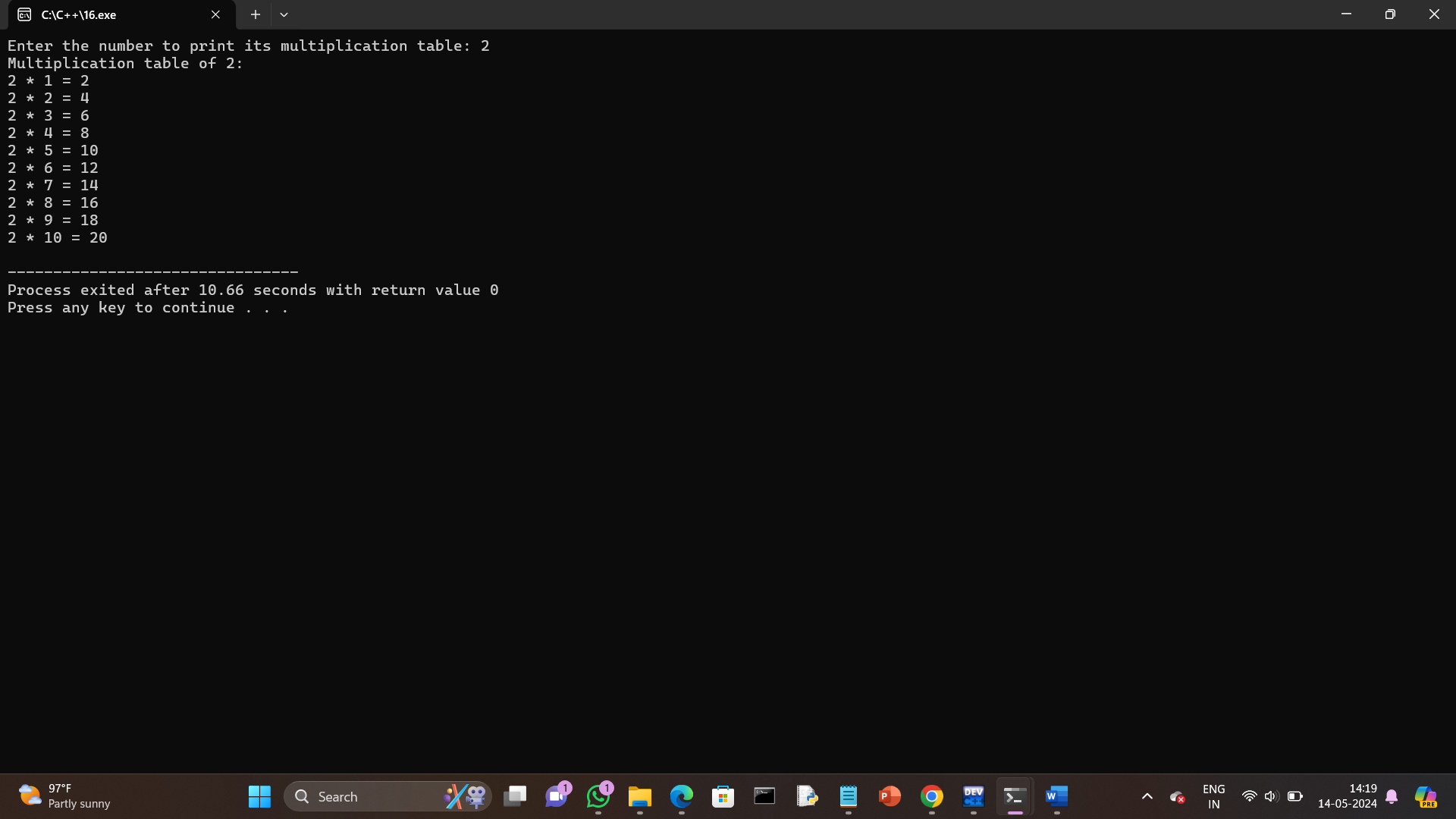
cout << num << " \* " << i << " = " << num \* i << endl;

}

return 0;

}

Output:



17.Fibonacci series using for loop.

Program:

#include <iostream>

using namespace std;

int main() {

int n;

cout << "Enter the number of terms for Fibonacci series: ";

cin >> n;

int first = 0, second = 1;

cout << "Fibonacci Series:" << endl;

cout << first << " " << second << " ";

for (int i = 2; i < n; ++i) {

int next = first + second;

cout << next << " ";

first = second;

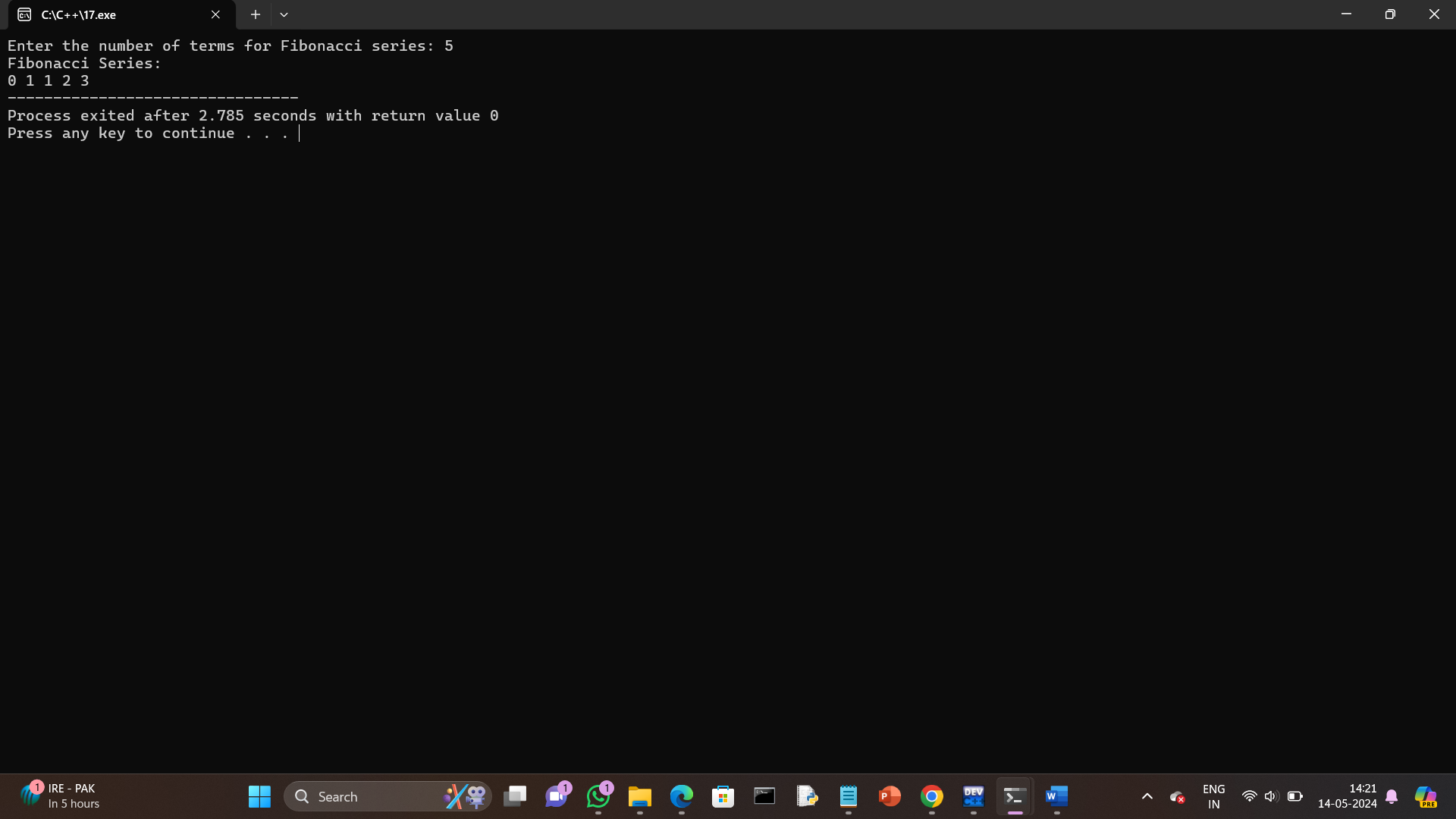
second = next;

}

return 0;

}

Output:



18.Prime number using for loop.

Program:

#include <iostream>

using namespace std;

int main() {

int num;

bool isPrime = true;

cout << "Enter a positive integer: ";

cin >> num;

// 0 and 1 are not prime numbers

if (num <= 1) {

isPrime = false;

} else {

// Check for divisibility from 2 to num/2

for (int i = 2; i <= num / 2; ++i) {

if (num % i == 0) {

isPrime = false;

break;

}

}

}

if (isPrime) {

cout << num << " is a prime number." << endl;

} else {

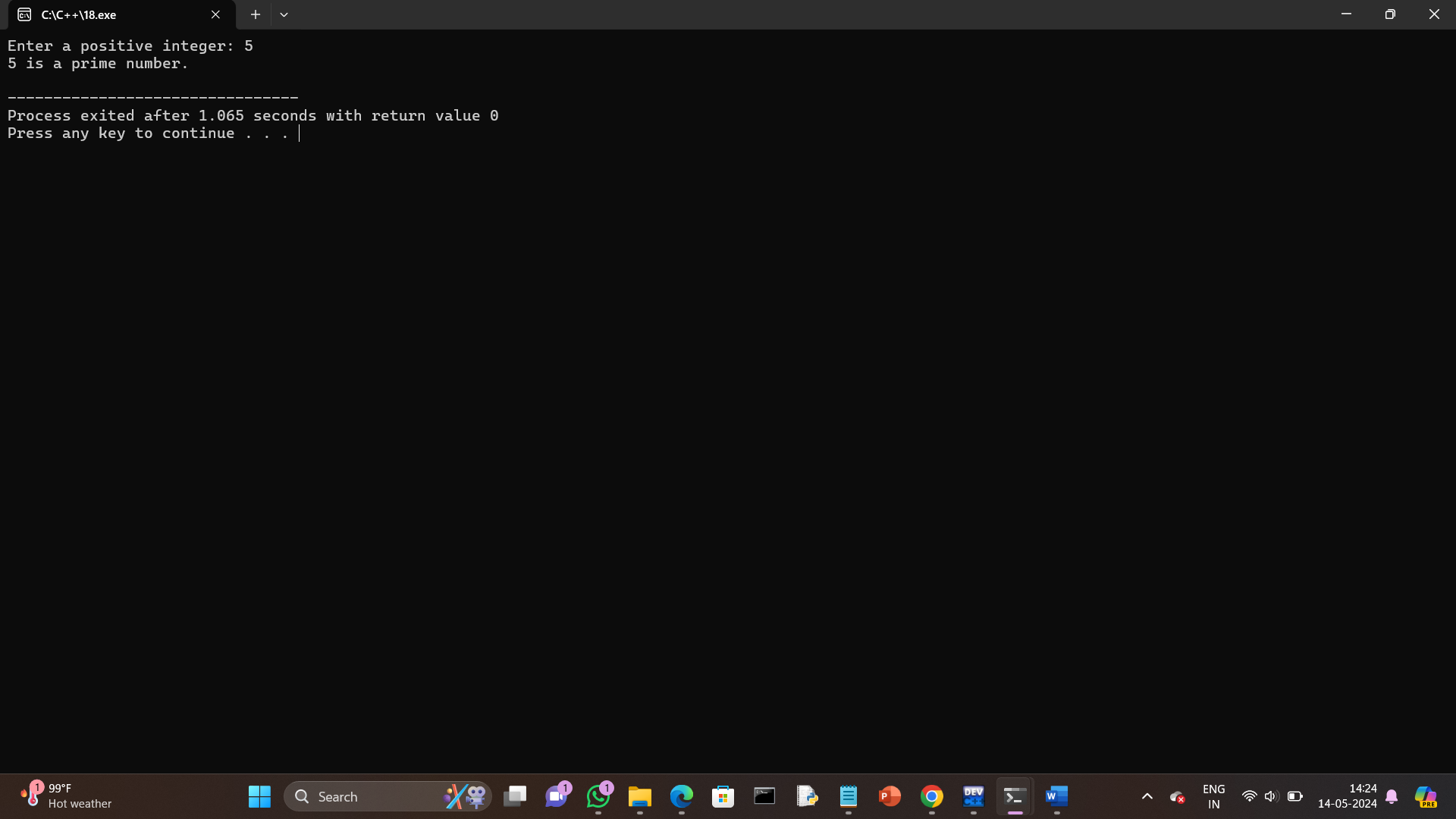
cout << num << " is not a prime number." << endl;

}

return 0;

}

Output:



19.Check the string is palindrome or not using while loop.

Program:

#include <iostream>

#include <string>

#include <cctype> // for std::tolower

using namespace std;

int main() {

string str;

bool isPalindrome = true;

cout << "Enter a string: ";

getline(cin, str);

// Convert the string to lowercase for case-insensitive comparison

for (size\_t i = 0; i < str.length(); ++i) {

str[i] = tolower(str[i]);

}

// Remove non-alphanumeric characters

string cleanStr;

for (size\_t i = 0; i < str.length(); ++i) {

if (isalnum(str[i])) {

cleanStr += str[i];

}

}

// Initialize start and end pointers

int start = 0;

int end = cleanStr.length() - 1;

// Check if the string is a palindrome using while loop

while (start < end) {

if (cleanStr[start] != cleanStr[end]) {

isPalindrome = false;

break;

}

++start;

--end;

}

if (isPalindrome) {

cout << "The string is a palindrome." << endl;

} else {

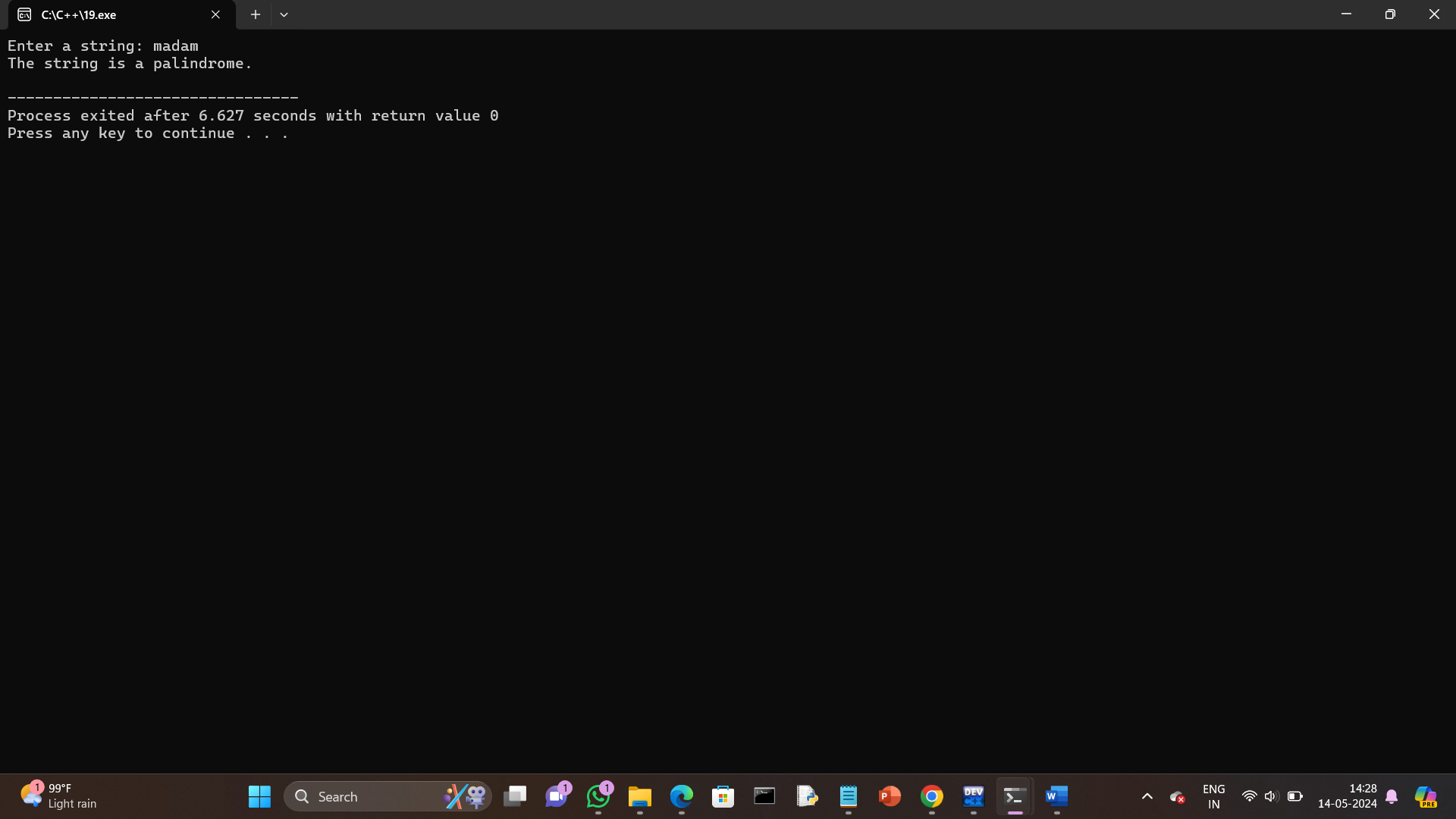
cout << "The string is not a palindrome." << endl;

}

return 0;

}

Output:



20.Sum of all digits using while loop (n=123 output:1+2+3=6).

Program:

#include <iostream>

using namespace std;

int main() {

int num, sum = 0, digit;

cout << "Enter a number: ";

cin >> num;

// Copy the number to a temporary variable

int temp = num;

// Calculate the sum of digits using a while loop

while (temp != 0) {

digit = temp % 10; // Extract the last digit

sum += digit; // Add the last digit to the sum

temp /= 10; // Remove the last digit from the number

}

cout << "Sum of all digits in " << num << " = " << sum << endl;

return 0;

}

Output:

