**1.Find factorial using function**

**Program:**

#include <iostream>

unsigned long long factorial\_recursive(int n) {

if (n == 0) {

return 1;

} else {

return n \* factorial\_recursive(n - 1);

}

}

int main() {

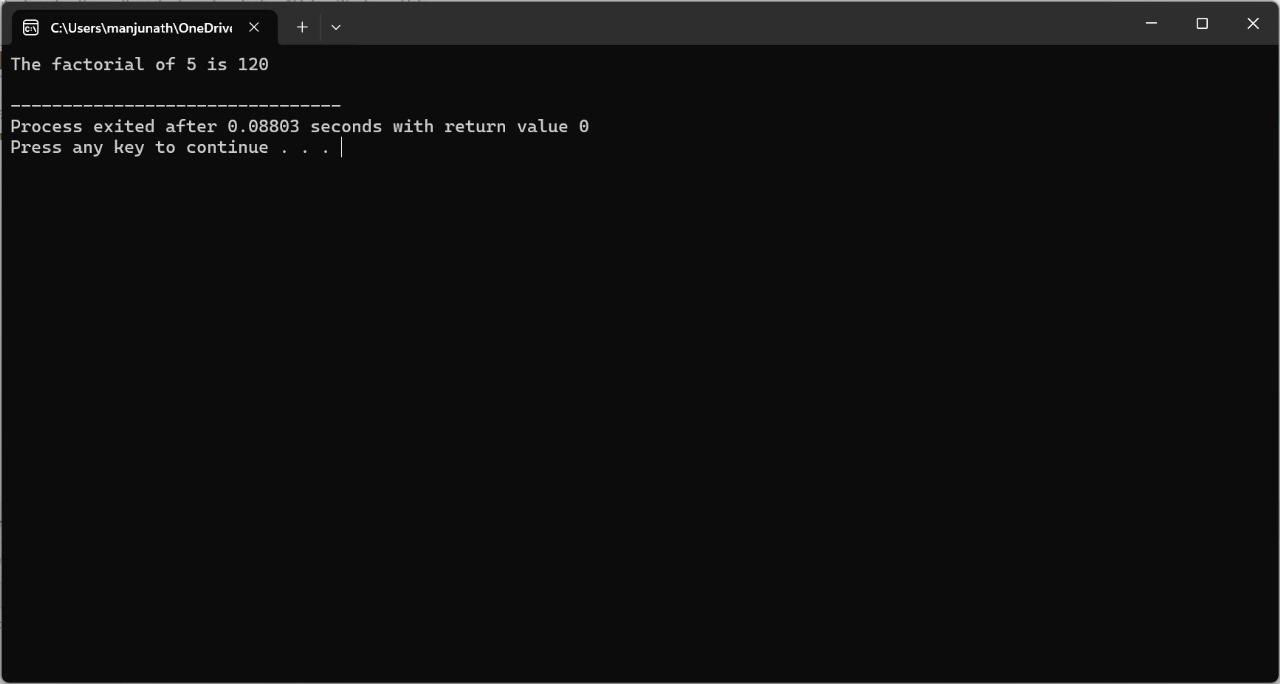
int num = 5;

std::cout << "The factorial of " << num << " is " << factorial\_recursive(num) << std::endl;

return 0;

}

**OUTPUT:**

****

**2. Find prime number using function**

**Program:**

#include <iostream>

#include <cmath>

bool is\_prime(int n) {

// Handle edge cases

if (n <= 1) {

return false;

}

if (n <= 3) {

return true;

}

if (n % 2 == 0 || n % 3 == 0) {

return false;

}

for (int i = 5; i \* i <= n; i += 6) {

if (n % i == 0 || n % (i + 2) == 0) {

return false;

}

}

return true;

}

int main() {

int num = 29; // Change this number to test other numbers

if (is\_prime(num)) {

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

} else {

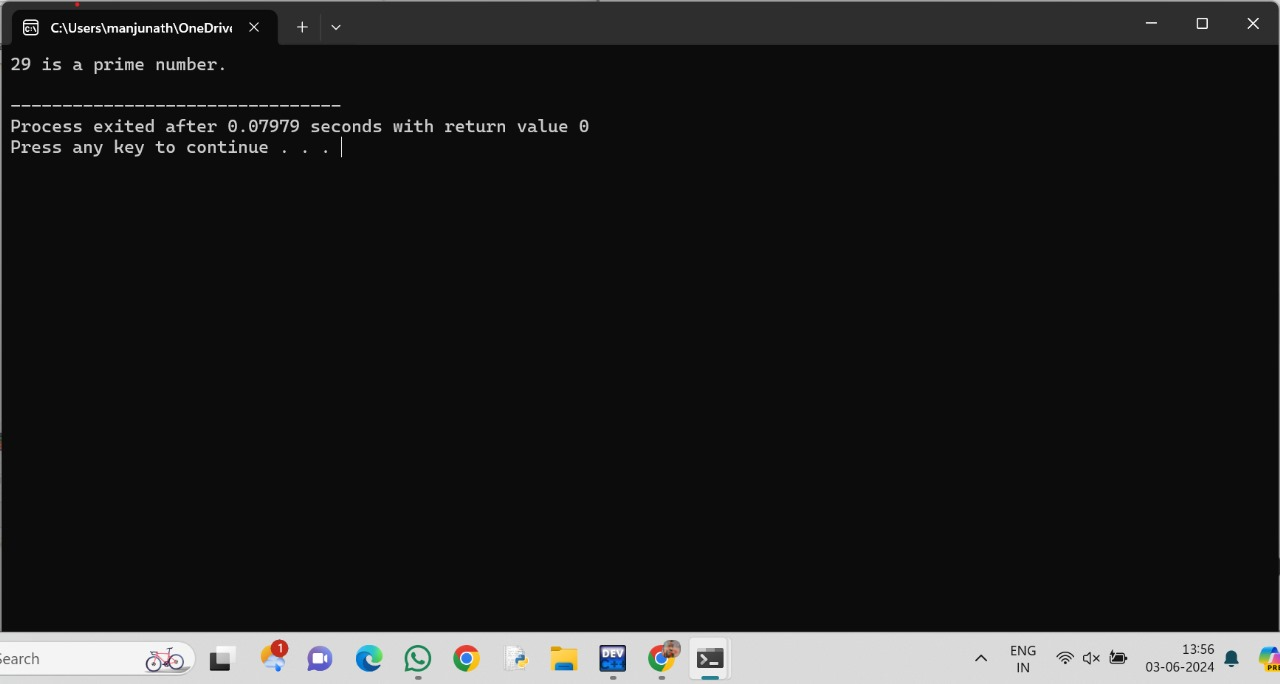
std::cout << num << " is not a prime number." << std::endl;

}

return 0;

}

**OUTPUT**

****

**3 Find the reverse of a string using function**

**Program:**

#include <iostream>

#include <string>

std::string reverse\_string(const std::string &str) {

std::string reversed\_str;

for (int i = str.size() - 1; i >= 0; --i) {

reversed\_str += str[i];

}

return reversed\_str;

}

int main() {

std::string original = "Hello, World!";

std::string reversed = reverse\_string(original);

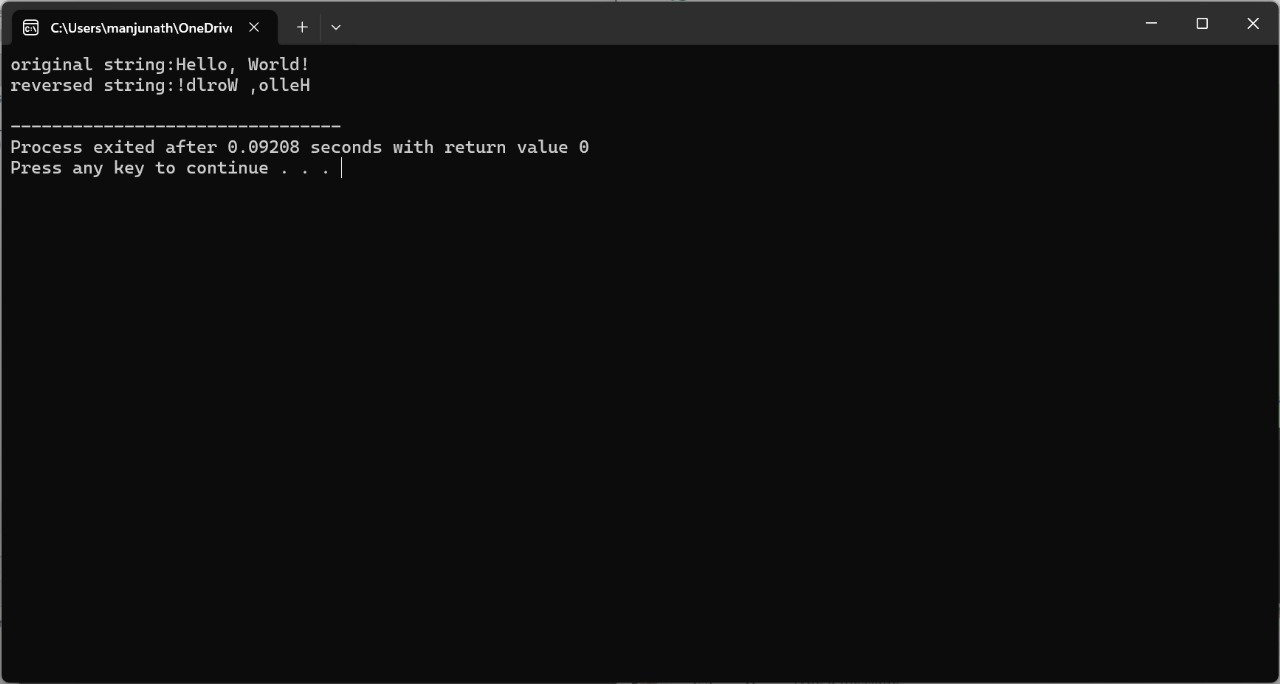
std::cout << "Original string: " << original << std::endl;

std::cout << "Reversed string: " << reversed << std::endl;

return 0;

}

**OUTPUT:**



**4. Find minimum and maximum element in an array using function**

#include <iostream>

#include <climits>

int find\_min(const int arr[], int size) {

int min\_element = INT\_MAX;

for (int i = 0; i < size; ++i) {

if (arr[i] < min\_element) {

min\_element = arr[i];

}

}

return min\_element;

}

int find\_max(const int arr[], int size) {

int max\_element = INT\_MIN;

for (int i = 0; i < size; ++i) {

if (arr[i] > max\_element) {

max\_element = arr[i];

}

}

return max\_element;

}

int main() {

int arr[] = {10, 20, 5, 7, 8, 25, 3};

int size = sizeof(arr) / sizeof(arr[0]);

int min\_element = find\_min(arr, size);

int max\_element = find\_max(arr, size);

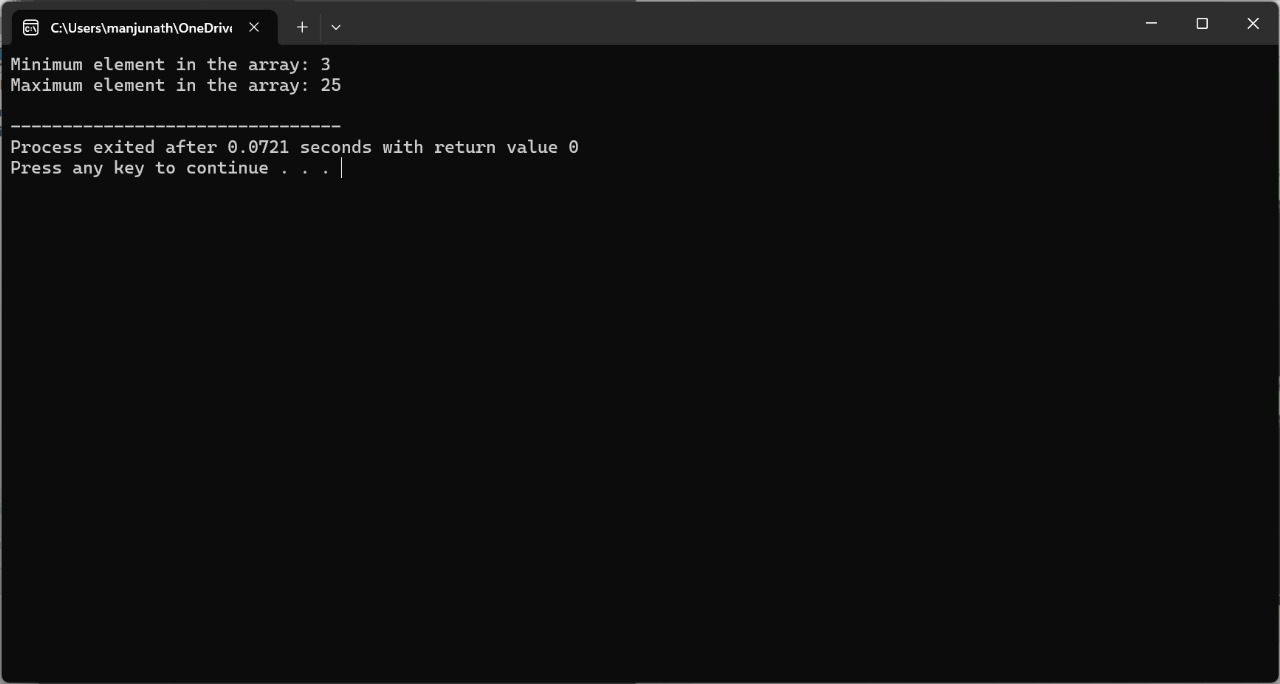
std::cout << "Minimum element in the array: " << min\_element << std::endl;

std::cout << "Maximum element in the array: " << max\_element << std::endl;

return 0;

}

**OUTPUT:**

****

**5. Find GCD of two number using function**

**Program:**

#include <iostream>

int gcd(int a, int b) {

while (b != 0) {

int temp = b;

b = a % b;

a = temp;

}

return a;

}

int main() {

int num1 = 56;

int num2 = 98;

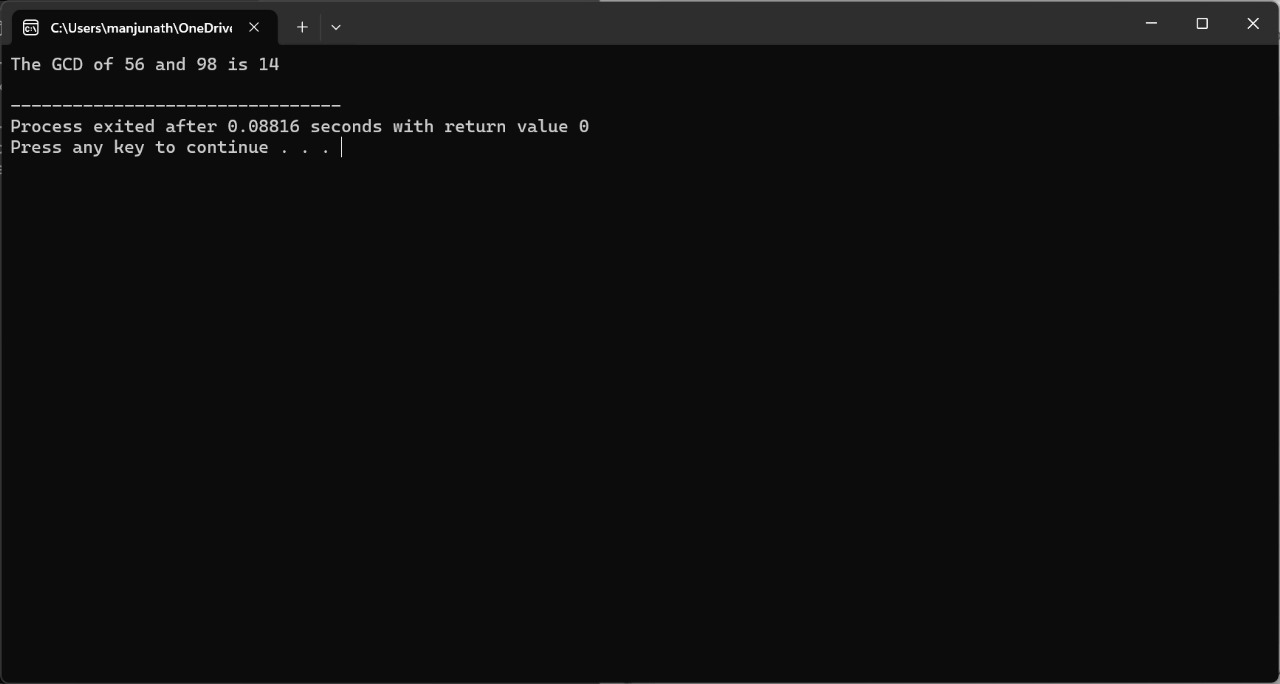
int result = gcd(num1, num2);

std::cout << "The GCD of " << num1 << " and " << num2 << " is " << result << std::endl;

return 0;

}

**OUTPUT:**

****

**6. Function to count the no of elements in a string**

**Program:**

#include <iostream>

#include <string>

int count\_characters(const std::string &str) {

return str.length();

}

int main() {

std::string myString = "Hello, World!";

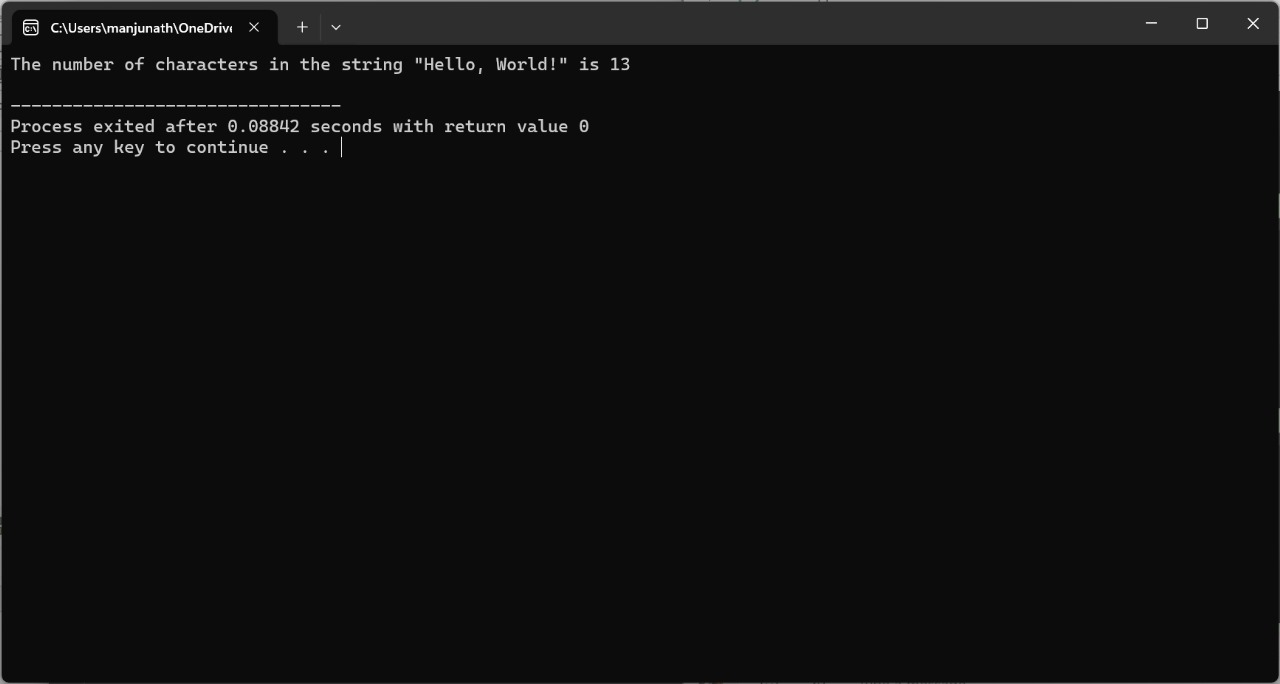
int count = count\_characters(myString);

std::cout << "The number of characters in the string \"" << myString << "\" is " << count << std::endl;

return 0;

}

**Output:**

****