Day -1

## Assignment

## Menda Mani Sai

## 192111399

# 1. Write a program to Print Fibonacci Series using recursion.

#include<stdio.h>

int fibbno(int n)

{

if(n==0)

return 0;

else if(n==1)

return 1;

else

return fibbno(n - 1) + fibbno(n- 2);

}

int main()

{

int n,i;

printf("Enter the range =");

scanf("%d", &n);

for(i=0;i<n;i++)

{

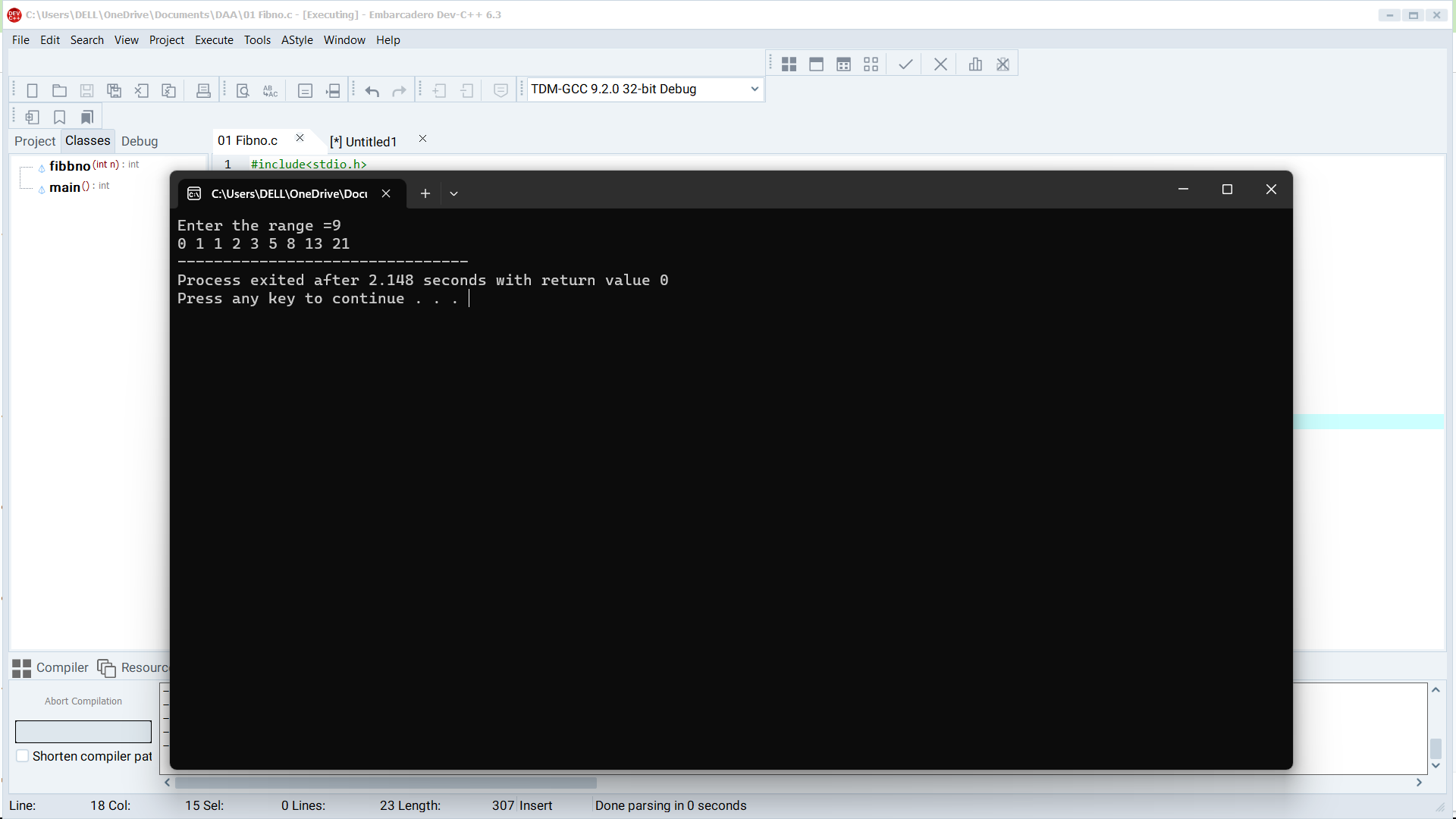
printf("%d",fibbno(i));

}

return 0;

}

# OUTPUT



# 2. Write a program to check the given no is Armstrong or not. .

#include<stdio.h>

#include<math.h>

int main()

{

int n,r,sum=0,temp;

printf("enter the number=");

scanf("%d",&n);

temp=n;

while(n>0)

{

r=n%10;

sum=sum+(r\*r\*r);

n=n/10;

}

if(temp==sum)

printf("armstrong number ");

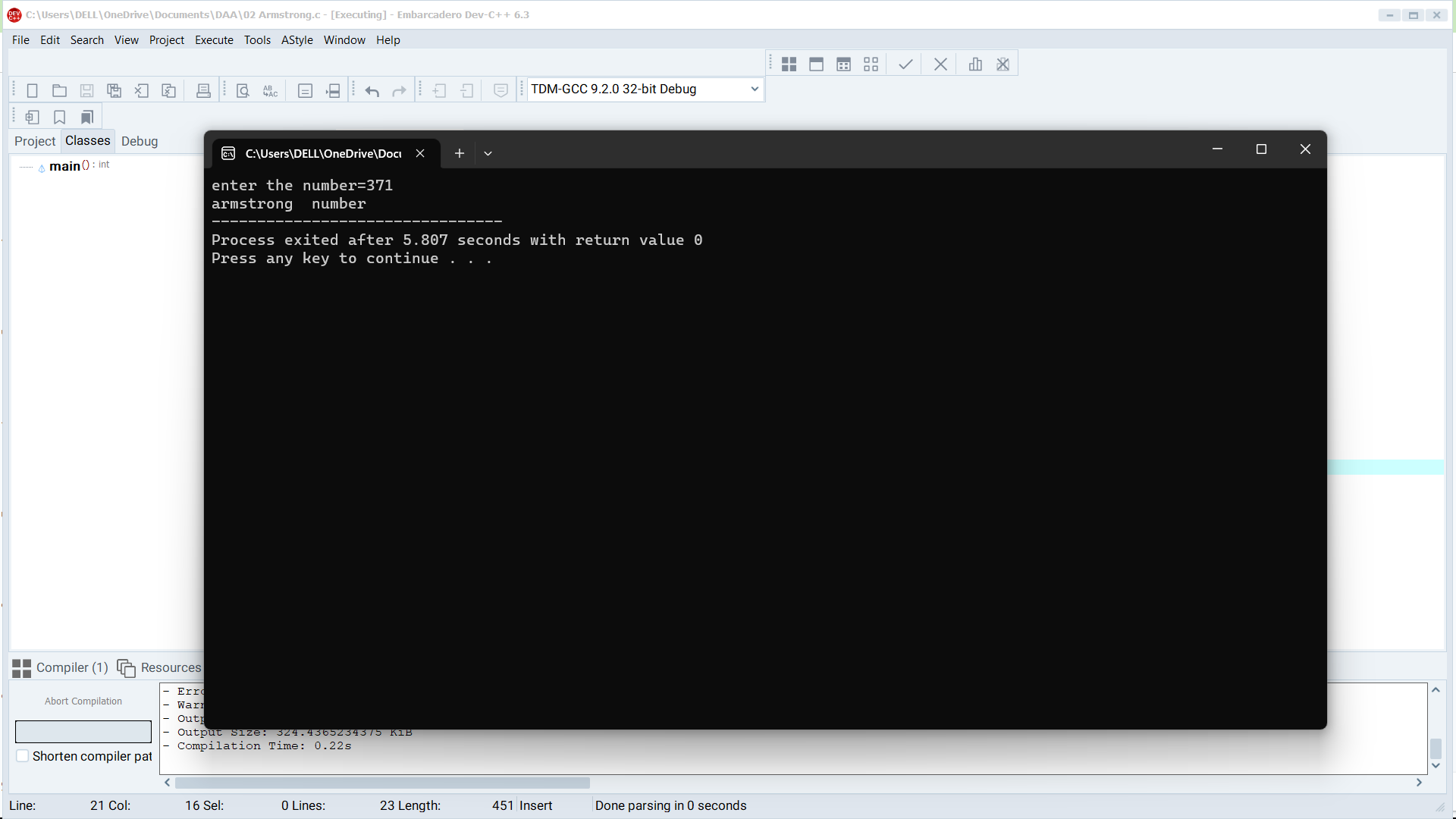
else

printf("not armstrong number");

return 0;

}

# OUTPUT



# 3. Write a program to find the GCD of two numbers.

#include <stdio.h>

int main()

{

int n1, n2, i, gcd;

printf("Enter two integers: ");

scanf("%d %d", &n1, &n2);

for(i=1; i <= n1 && i <= n2; ++i)

{

if(n1%i==0 && n2%i==0)

gcd = i;

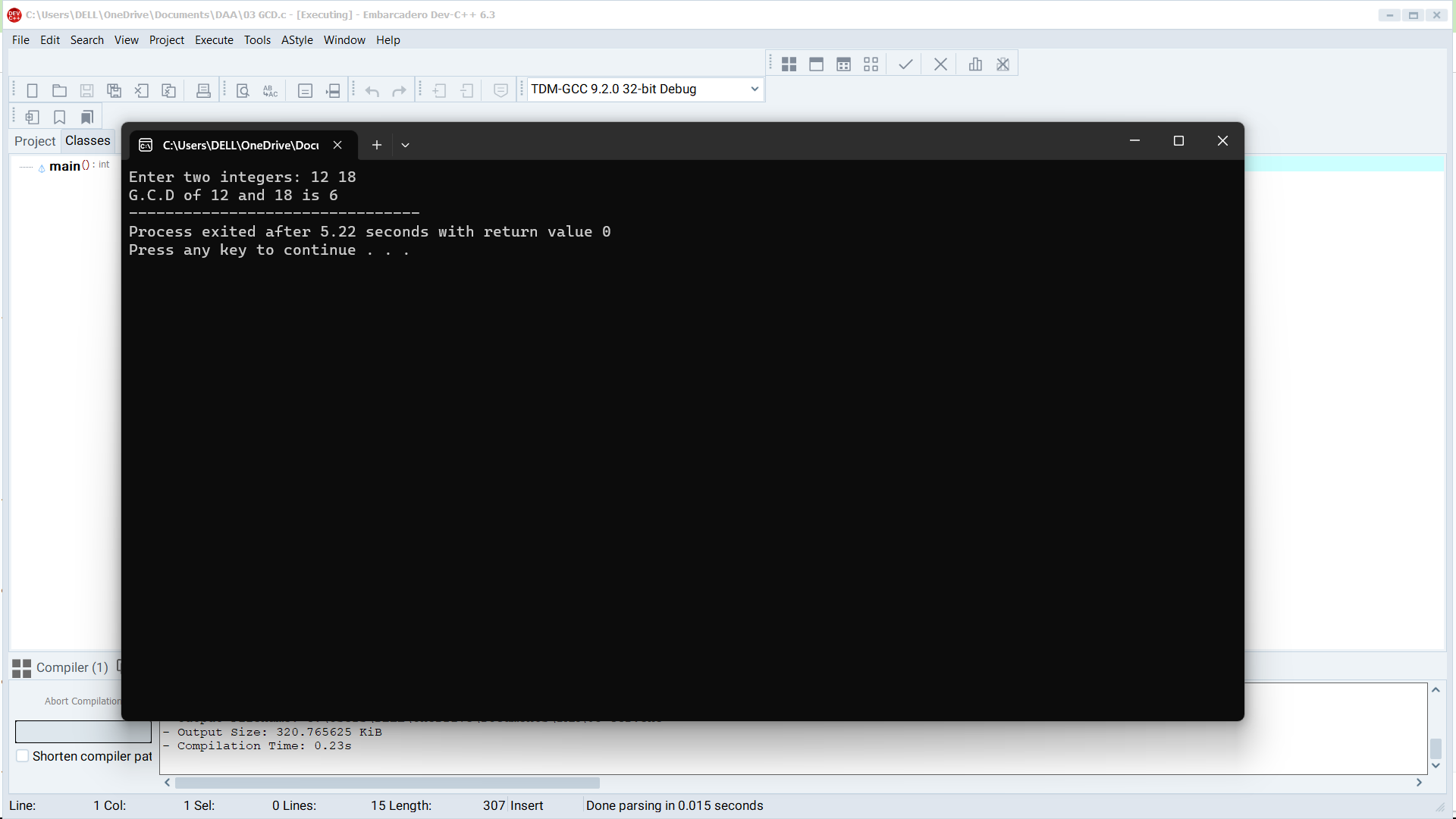
}

printf("G.C.D of %d and %d is %d", n1, n2, gcd);

return 0;

}

# OUTPUT



# 4. Write a program to get the largest element of an array.

#include<stdio.h>

int main()

{

int a[1000];

int i,n;

printf("Enter the size of the array= ");

scanf("%d",&n);

for (i=0;i<n;i++)

{

scanf("%d", &a[i]);

}

for (i=0;i<n;i++){

printf("%d", a[i]);

}

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

if (a[0] < a[i]) {

a[0] = a[i];

}

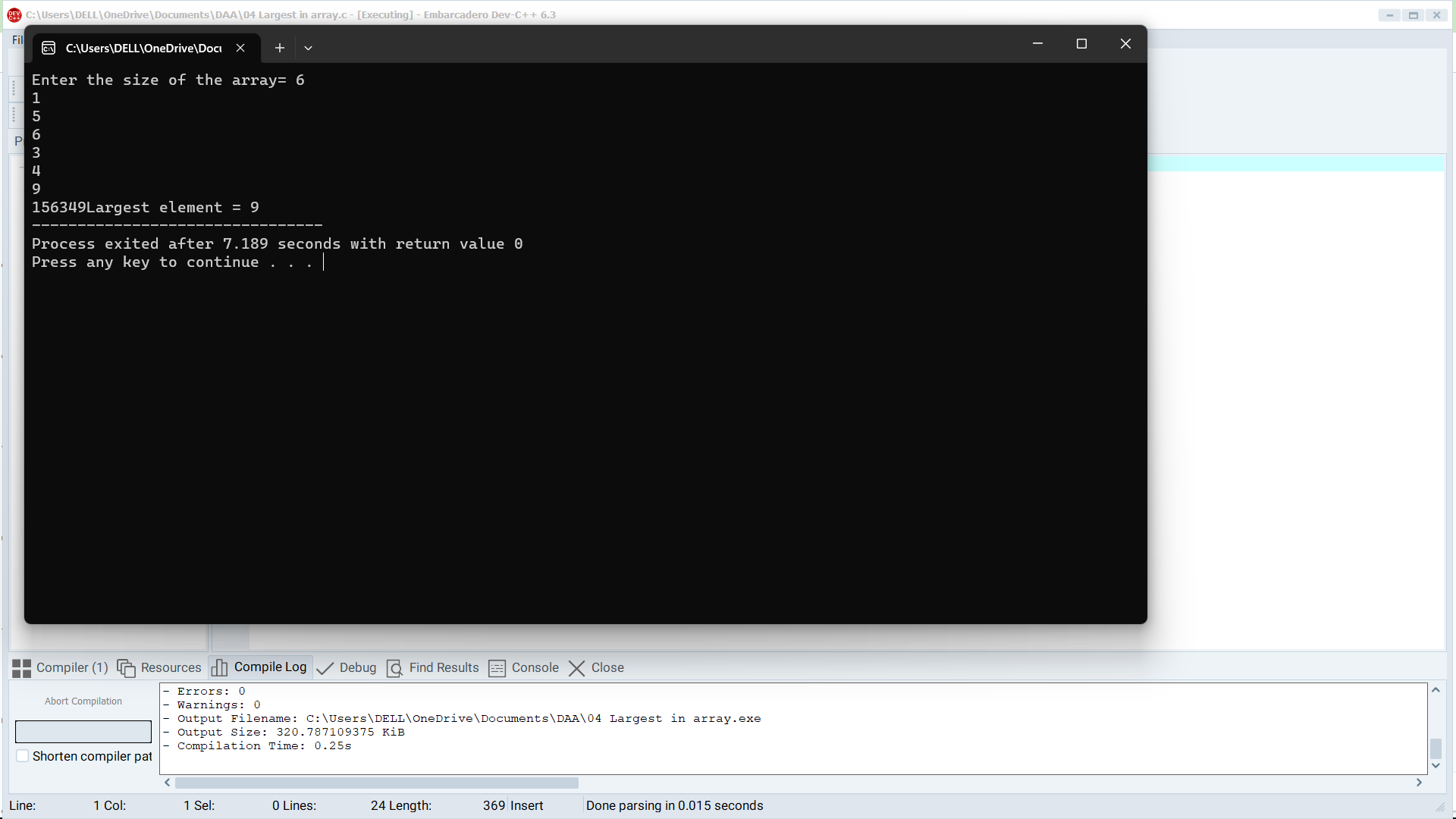
}

printf("Largest element = %d", a[0]);

return 0;

}

# OUTPUT



# 5. Write a program to find the Factorial of a number.

#include<stdio.h>

int fact(int a)

{

if(a==0)

return 1;

else if(a==1)

return 1;

else

return (a\*fact(a-1));

}

int main()

{

int a,i;

printf("Enter the number to know the factorial= ");

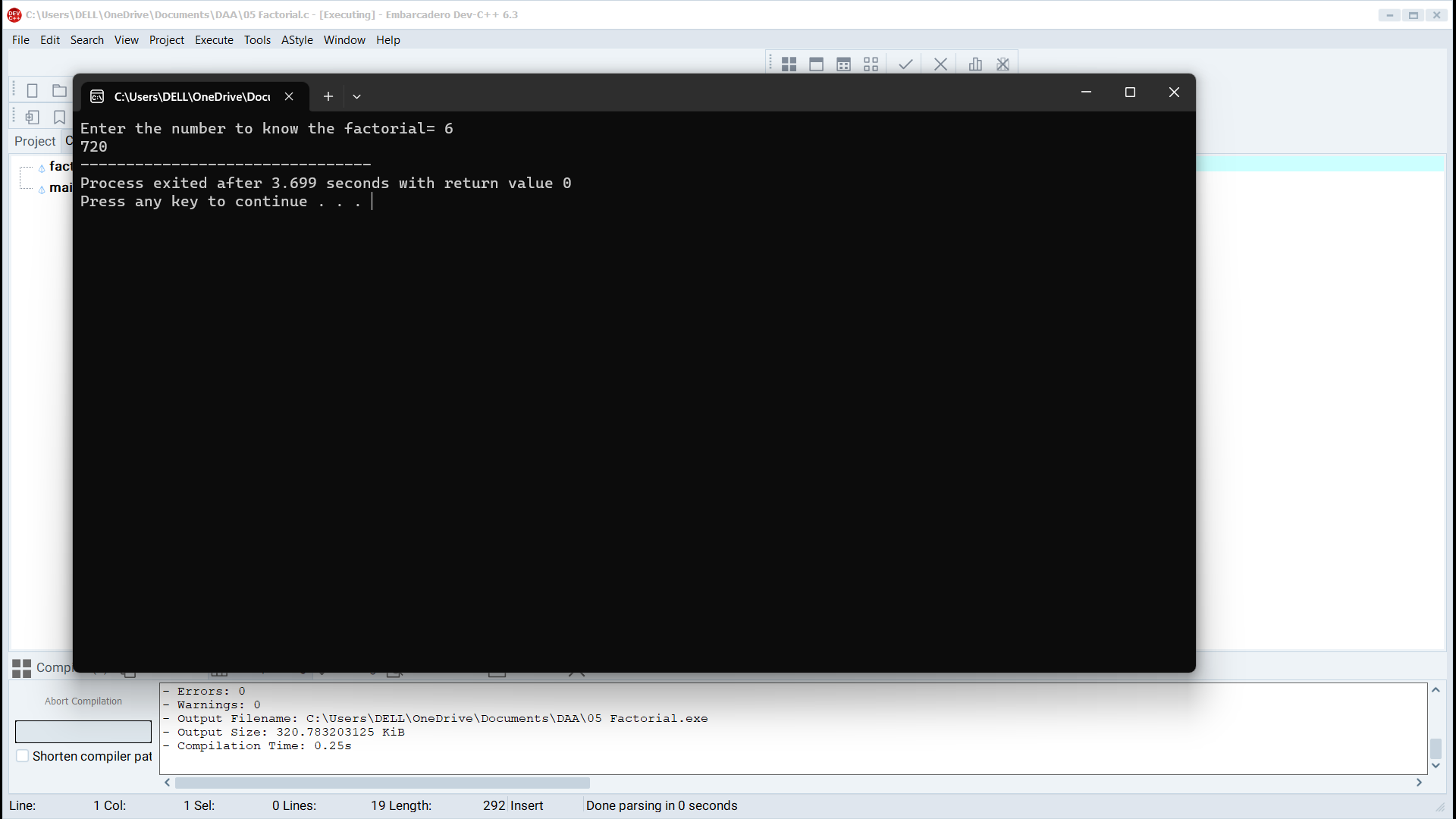
scanf("%d", &a);

printf("%d", fact(a));

return 0;

}

# OUTPUT



# 6. Write a program to check a number is a prime number or not.

#include<stdio.h>

int main()

{

int a,c=0;

printf("Enter a number= ");

scanf("%d", &a);

for(int i=1;i<=a;i++)

{

if(a%i==0){

c++;

}

}

if(c==2)

printf("The number %d is prime",a);

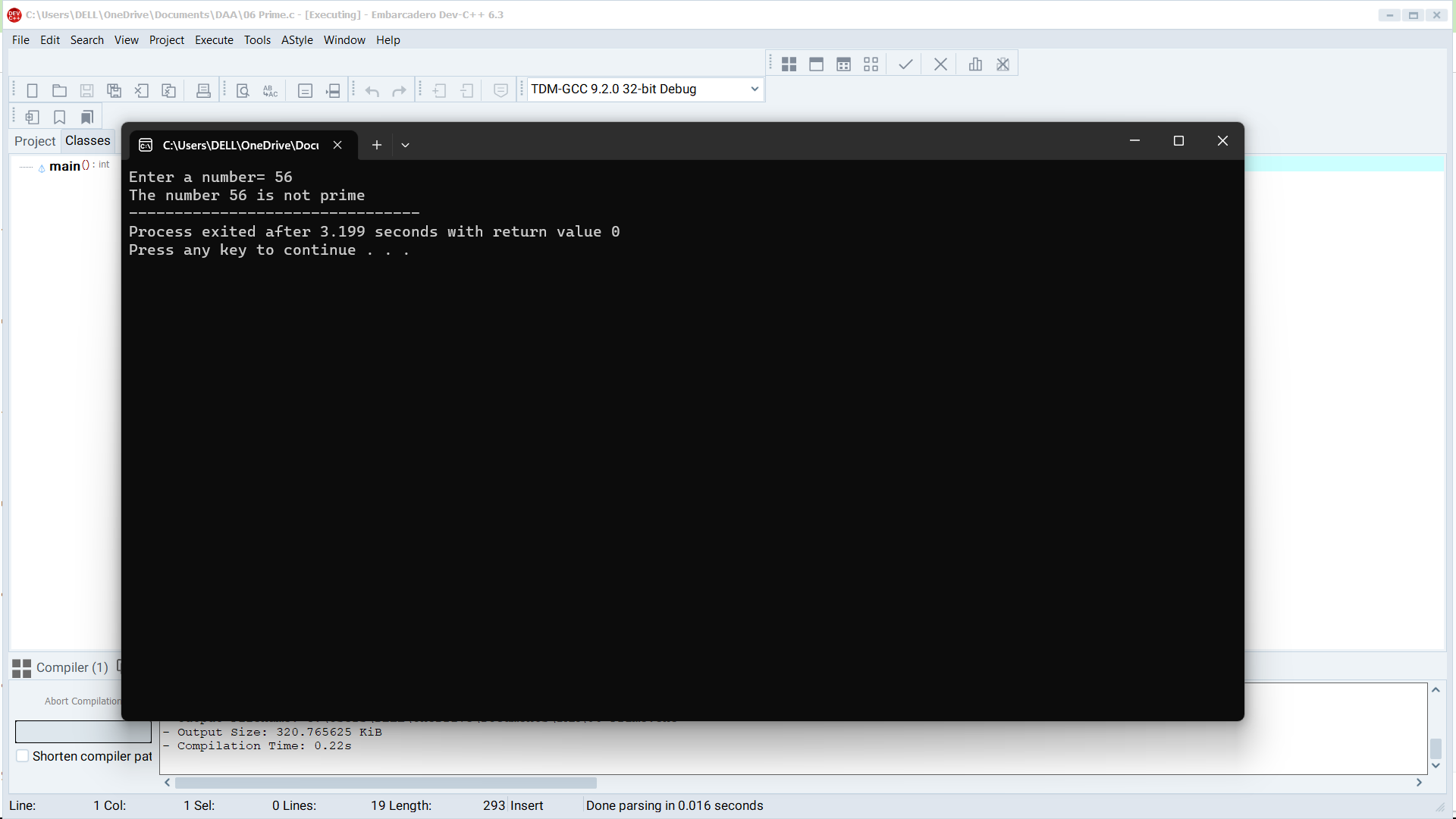
else

printf("The number %d is not prime",a);

return 0;

}

# OUTPUT



# 7. Write a program to perform Selection sort.

#include <stdio.h>

void selectionSort(int arr[], int n) {

for (int i = 0; i < n - 1; i++) {

int minIndex = i;

for (int j = i + 1; j < n; j++) {

if (arr[j] < arr[minIndex]) {

minIndex = j;

}

}

int temp = arr[minIndex];

arr[minIndex] = arr[i];

arr[i] = temp;

}

}

int main() {

int n;

printf("Enter the number of elements: ");

scanf("%d", &n);

int arr[n];

printf("Enter the elements:\n");

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

scanf("%d", &arr[i]);

}

selectionSort(arr, n);

printf("Sorted array:\n");

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

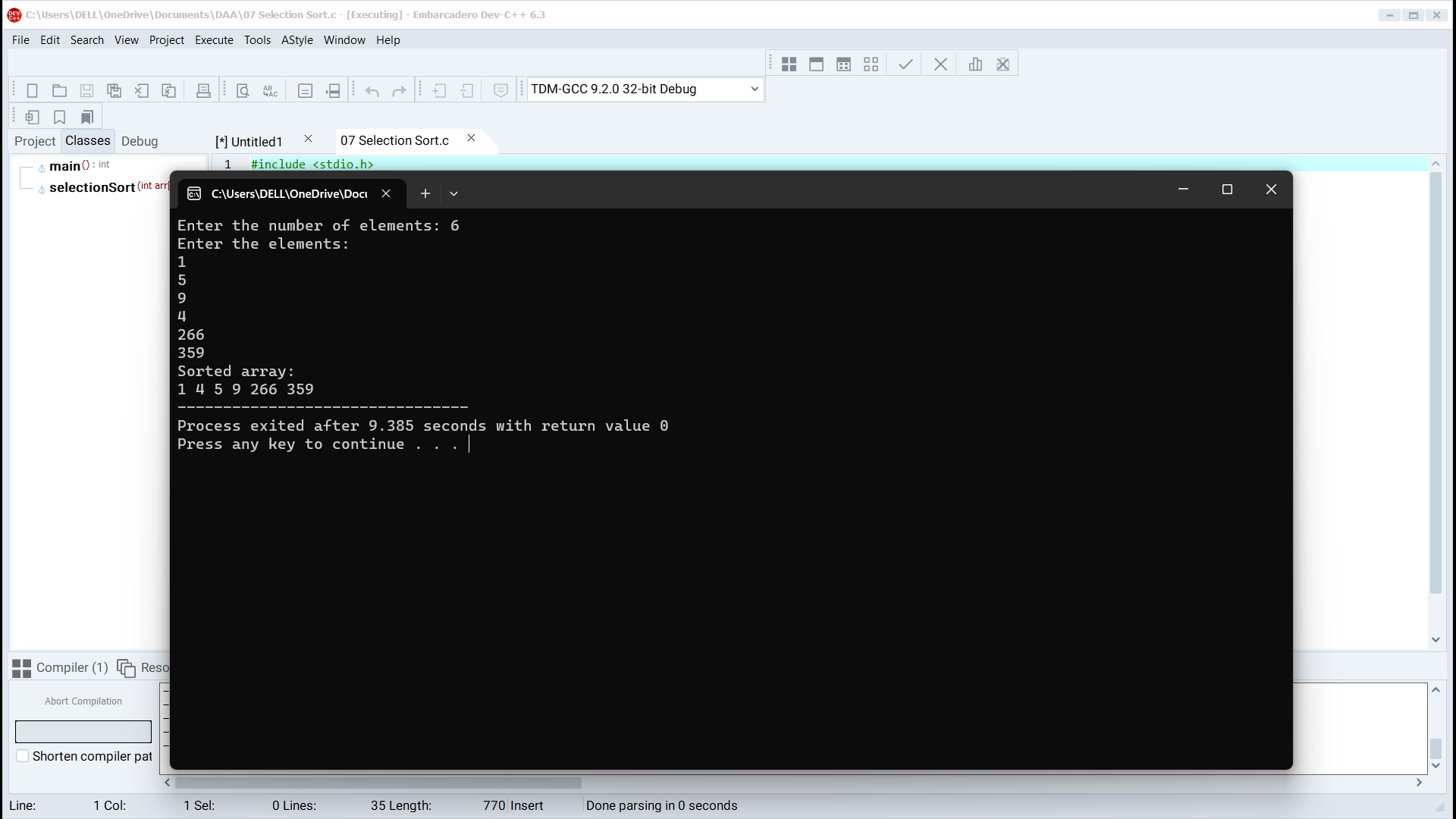
printf("%d ", arr[i]);

}

return 0;

}

# OUTPUT



# 8. Write a program to perform Bubble sort

#include <stdio.h>

void bubbleSort(int arr[], int n) {

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - i - 1; j++) {

if (arr[j] > arr[j + 1]) {

int temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

int main() {

int n;

printf("Enter the number of elements: ");

scanf("%d", &n);

int arr[n];

printf("Enter the elements:\n");

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

scanf("%d", &arr[i]);

}

bubbleSort(arr, n);

printf("Sorted array:\n");

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

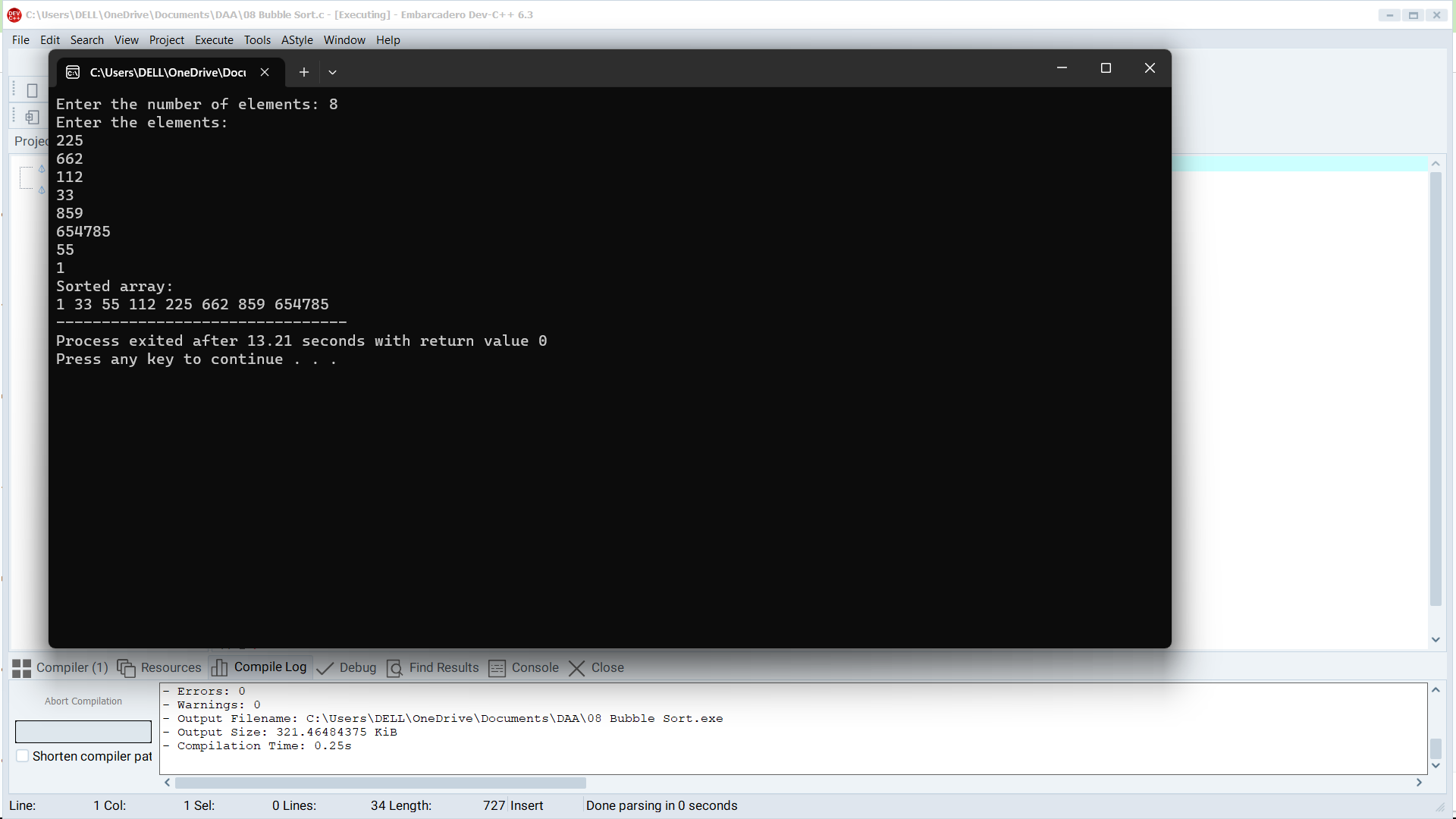
printf("%d ", arr[i]);

}

return 0;

}

# OUTPUT



# 9. Write a program for to multiply two Matrixes

#include <stdio.h>

void matrix\_multiply(int a[][100], int b[][100], int result[][100], int rows\_a, int cols\_a, int cols\_b) {

for (int i = 0; i < rows\_a; i++) {

for (int j = 0; j < cols\_b; j++) {

result[i][j] = 0;

for (int k = 0; k < cols\_a; k++) {

result[i][j] += a[i][k] \* b[k][j];

}

}

}

}

void print\_matrix(int matrix[][100], int rows, int cols) {

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

for (int j = 0; j < cols; j++) {

printf("%d ", matrix[i][j]);

}

printf("\n");

}

}

int main() {

int rows\_a, cols\_a, rows\_b, cols\_b;

printf("Enter the dimensions of matrix A (rows columns): ");

scanf("%d %d", &rows\_a, &cols\_a);

printf("Enter the dimensions of matrix B (rows columns): ");

scanf("%d %d", &rows\_b, &cols\_b);

if (cols\_a != rows\_b) {

printf("Matrix multiplication not possible due to incompatible dimensions.\n");

return 1;

}

int matrixA[100][100];

int matrixB[100][100];

printf("Enter the elements of matrix A:\n");

for (int i = 0; i < rows\_a; i++) {

for (int j = 0; j < cols\_a; j++) {

scanf("%d", &matrixA[i][j]);

}

}

printf("Enter the elements of matrix B:\n");

for (int i = 0; i < rows\_b; i++) {

for (int j = 0; j < cols\_b; j++) {

scanf("%d", &matrixB[i][j]);

}

}

int resultMatrix[100][100];

matrix\_multiply(matrixA, matrixB, resultMatrix, rows\_a, cols\_a, cols\_b);

printf("\nMatrix A:\n");

print\_matrix(matrixA, rows\_a, cols\_a);

printf("\nMatrix B:\n");

print\_matrix(matrixB, rows\_b, cols\_b);

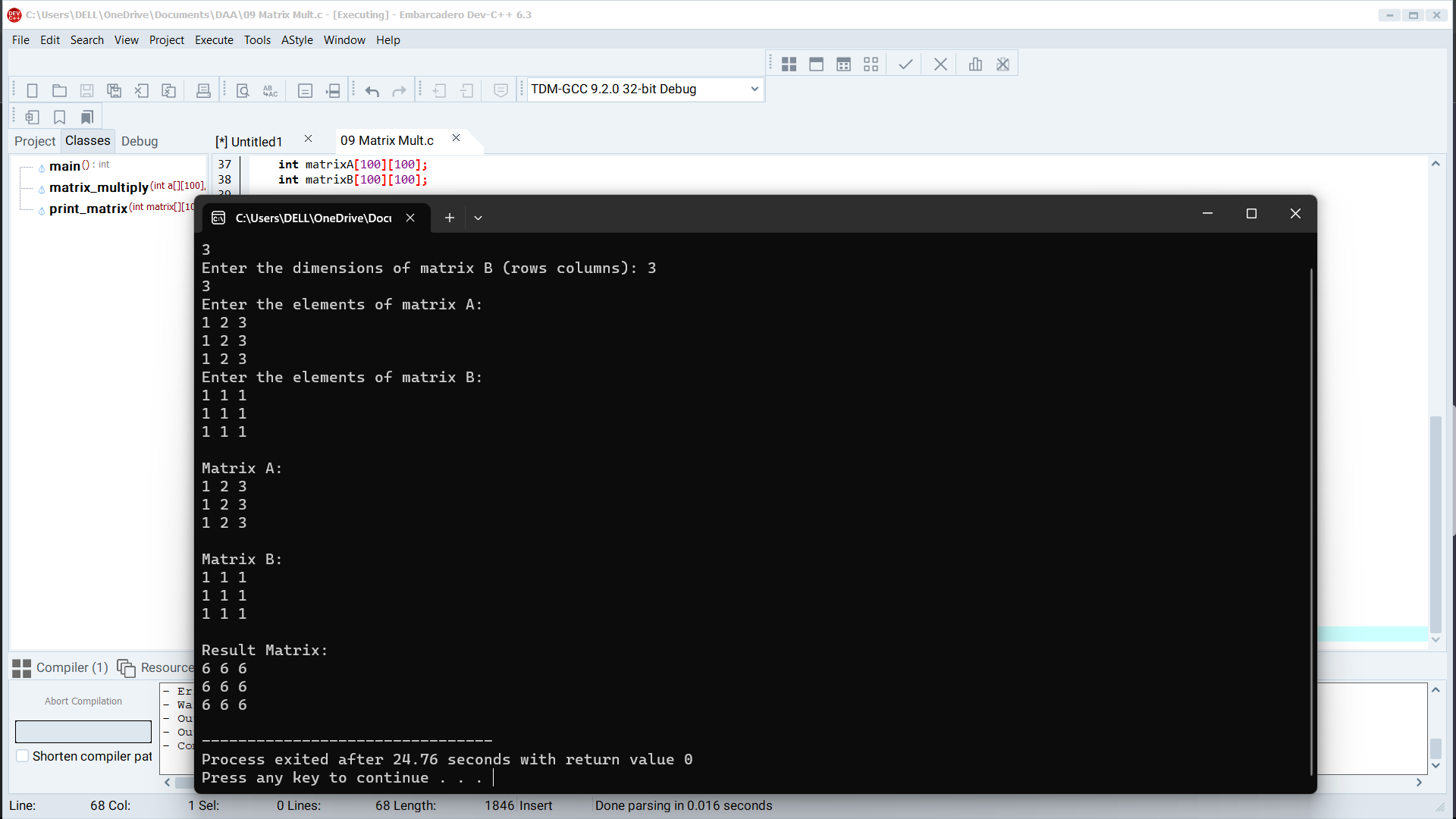
printf("\nResult Matrix:\n");

print\_matrix(resultMatrix, rows\_a, cols\_b);

return 0;

}

# OUTPUT



# 10. Write a program for to check whether a given String is Palindrome or not

#include <stdio.h>

#include <stdbool.h>

#include <string.h>

bool isPalindrome(const char str[]) {

int length = strlen(str);

for (int i = 0; i < length / 2; i++) {

if (str[i] != str[length - 1 - i]) {

return false;

}

}

return true;

}

int main() {

char str[100];

printf("Enter a string: ");

scanf("%s", str);

if (isPalindrome(str)) {

printf("The string is a palindrome.\n");

} else {

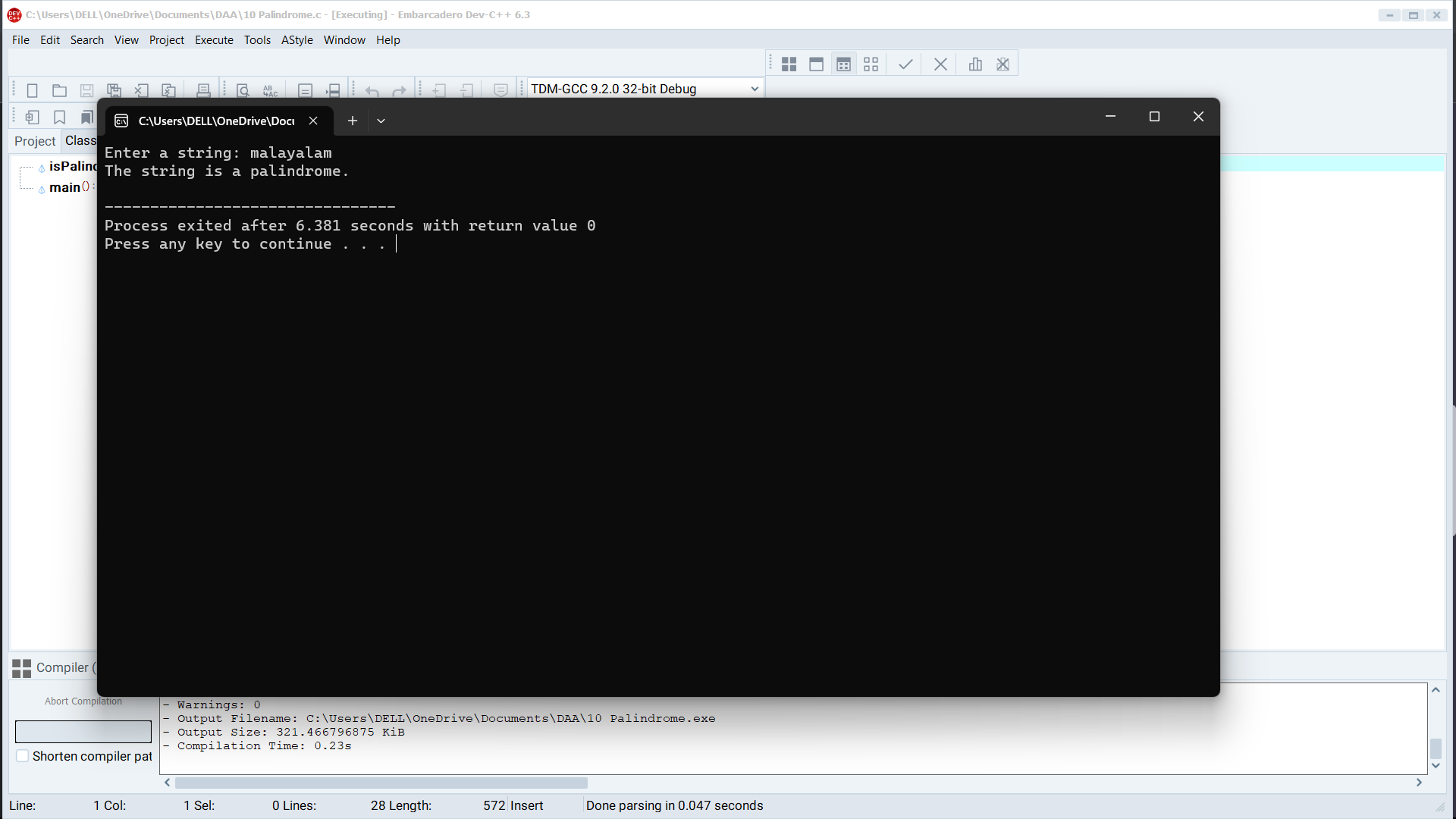
printf("The string is not a palindrome.\n");

}

return 0;

}

# OUTPUT



# 11. Write a program for to copy one string to another

#include<stdio.h>

#include<conio.h>

int main()

{

char str1[20], str2[20];

printf("Enter the string: ");

scanf("%s", str1);

printf("\nString 1 = %s", str1);

strcpy(str2, str1);

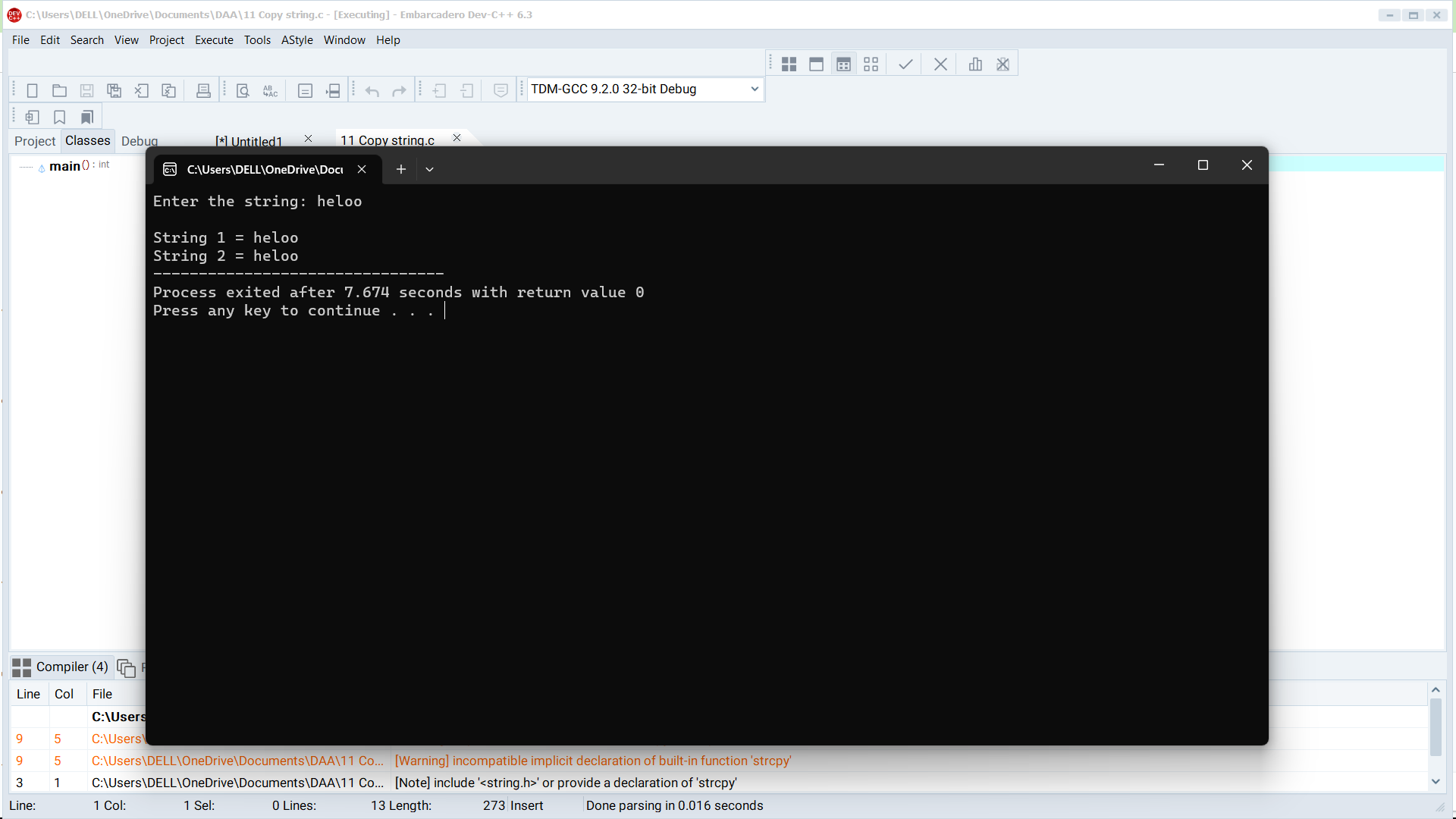
printf("\nString 2 = %s", str2);

getch();

return 0;

}

# OUTPUT



# 12. Write a Program to perform binary search.

#include <stdio.h>

int binarySearch(int arr[], int n, int target) {

int left = 0;

int right = n - 1;

while (left <= right) {

int mid = left + (right - left) / 2;

if (arr[mid] == target) {

return mid; // Element found at index 'mid'

} else if (arr[mid] < target) {

left = mid + 1; // Search the right half

} else {

right = mid - 1; // Search the left half

}

}

return -1; // Element not found

}

int main() {

int n;

printf("Enter the number of elements: ");

scanf("%d", &n);

int arr[n];

printf("Enter the elements in sorted order:\n");

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

scanf("%d", &arr[i]);

}

int target;

printf("Enter the element to search for: ");

scanf("%d", &target);

int index = binarySearch(arr, n, target);

if (index != -1) {

printf("Element found at index %d\n", index);

} else {

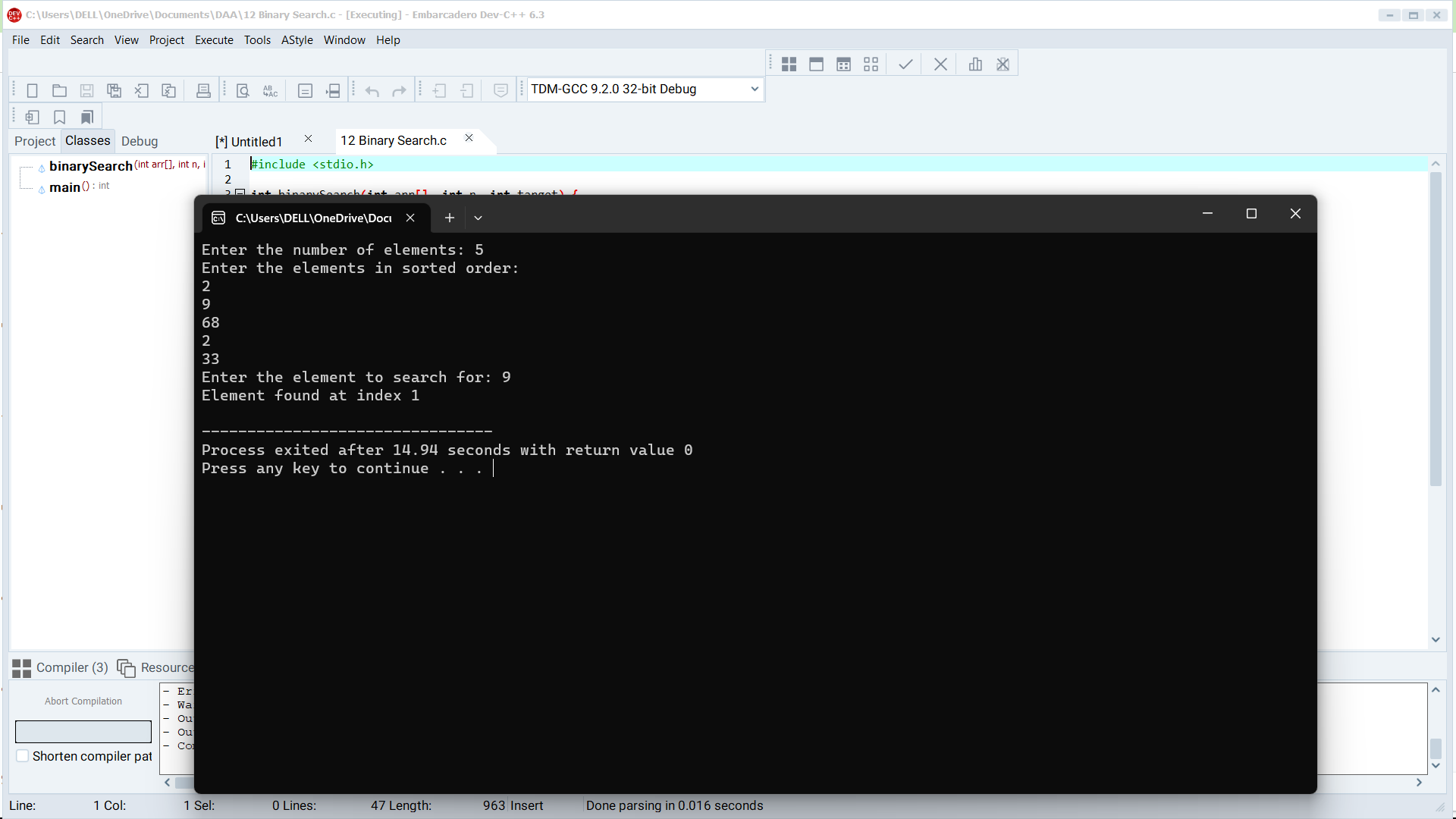
printf("Element not found\n");

}

return 0;

}

# OUTPUT



# 13. Write a program to print the reverse of a string

#include <stdio.h>

#include <string.h>

int main()

{

char Str[1000];

int i;

printf("Enter the String: ");

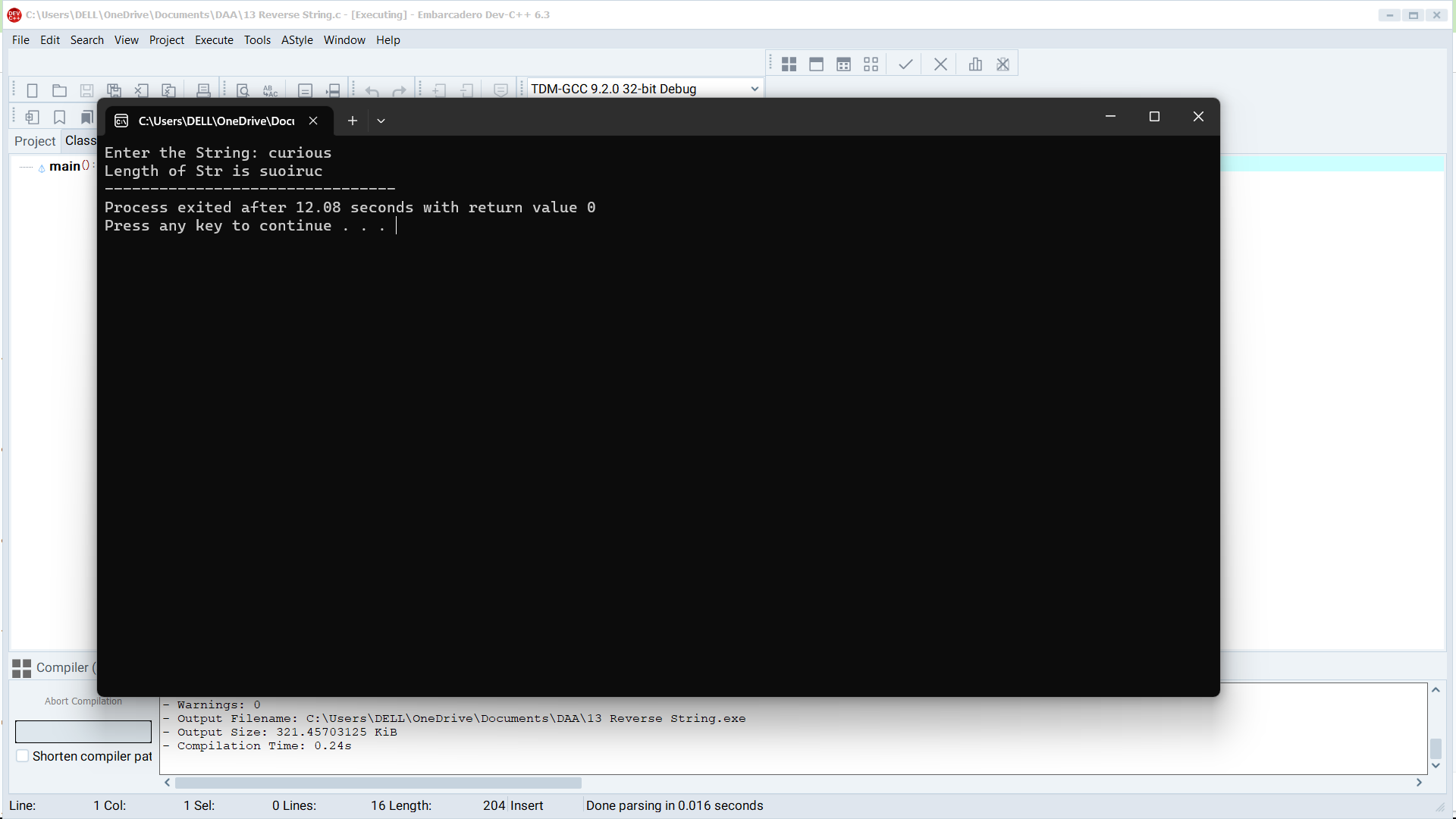
scanf("%s", Str);

printf("Length of Str is %s", strrev(Str));

return 0;

}

# OUTPUT



# 14. Write a program to find the length of a string.

#include <stdio.h>

#include <string.h>

int main()

{

char Str[1000];

int i;

printf("Enter the String: ");

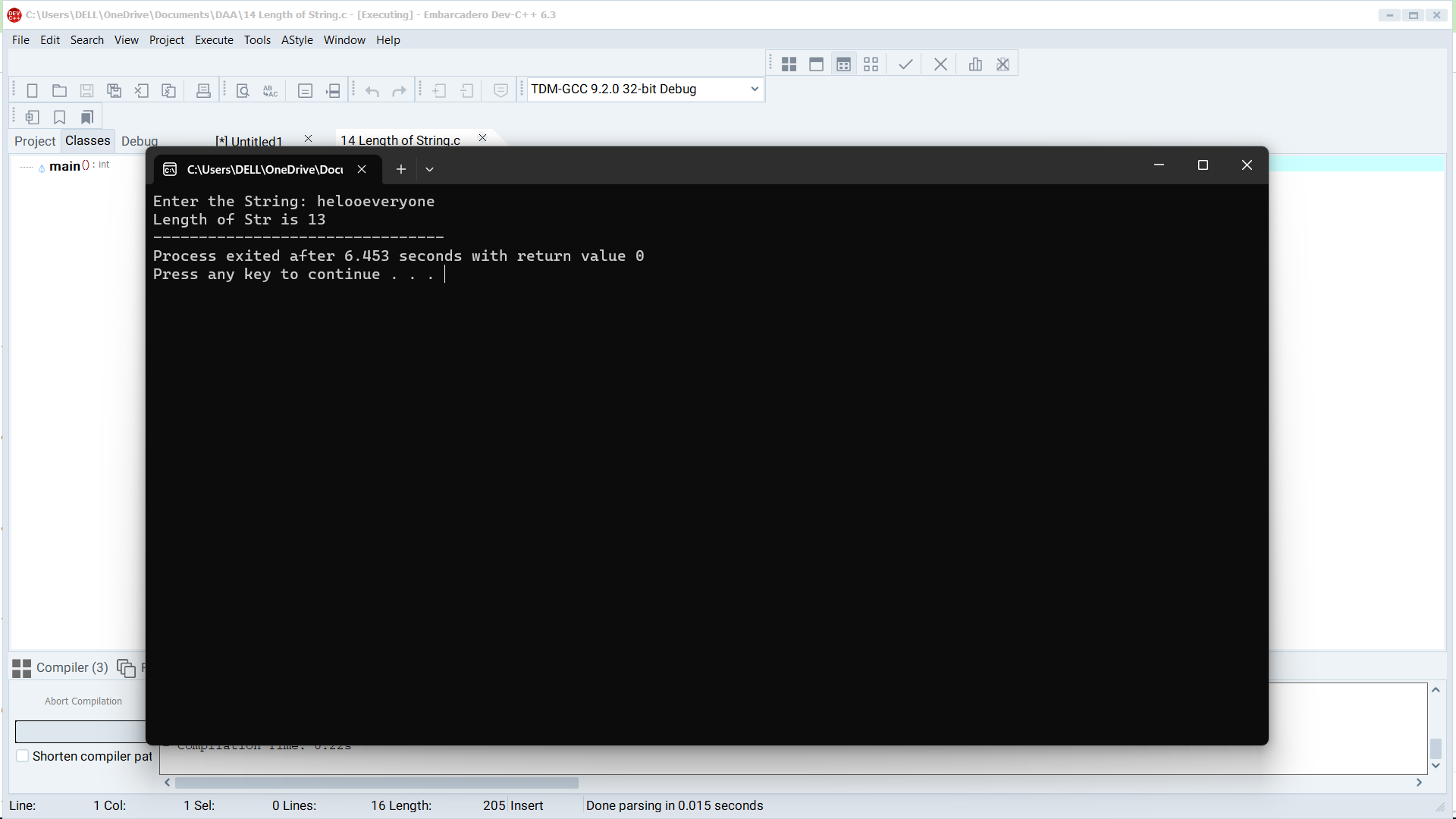
scanf("%s", Str);

printf("Length of Str is %ld", strlen(Str));

return 0;

}

# OUTPUT



# 15. Write a program to perform Strassen’s Matrix Multiplication.

#include<stdio.h>

int main(){

int a[2][2], b[2][2], c[2][2], i, j;

int m1, m2, m3, m4 , m5, m6, m7;

printf("Enter the 4 elements of first matrix: ");

for(i = 0;i < 2; i++)

for(j = 0;j < 2; j++)

scanf("%d", &a[i][j]);

printf("Enter the 4 elements of second matrix: ");

for(i = 0; i < 2; i++)

for(j = 0;j < 2; j++)

scanf("%d", &b[i][j]);

printf("\nThe first matrix is\n");

for(i = 0; i < 2; i++){

printf("\n");

for(j = 0; j < 2; j++)

printf("%d\t", a[i][j]);

}

printf("\nThe second matrix is\n");

for(i = 0;i < 2; i++){

printf("\n");

for(j = 0;j < 2; j++)

printf("%d\t", b[i][j]);

}

m1= (a[0][0] + a[1][1]) \* (b[0][0] + b[1][1]);

m2= (a[1][0] + a[1][1]) \* b[0][0];

m3= a[0][0] \* (b[0][1] - b[1][1]);

m4= a[1][1] \* (b[1][0] - b[0][0]);

m5= (a[0][0] + a[0][1]) \* b[1][1];

m6= (a[1][0] - a[0][0]) \* (b[0][0]+b[0][1]);

m7= (a[0][1] - a[1][1]) \* (b[1][0]+b[1][1]);

c[0][0] = m1 + m4- m5 + m7;

c[0][1] = m3 + m5;

c[1][0] = m2 + m4;

c[1][1] = m1 - m2 + m3 + m6;

printf("\nAfter multiplication using Strassen's algorithm \n");

for(i = 0; i < 2 ; i++){

printf("\n");

for(j = 0;j < 2; j++)

printf("%d\t", c[i][j]);

}

return 0;

}

# OUTPUT

