|  |  |
| --- | --- |
| download | COMSATS University Islamabad, Vehari Campus Department of Computer Science |

**Class: BCS-SP22-4A Submission Deadline: 10 Sep 2023**

**Subject: Data Structures and Algorithms-Lab**

**Name: Muhammad Abdullah**

**Instructor: Yasmeen Jana Max Marks: 10 Reg. No: SP22-BCS-036**

**Email:** [**yasmeenjana@cuivehari.edu.pk**](mailto:yasmeenjana@cuivehari.edu.pk)

**You can ask queries related to Lab Activities on the above email.**

**Activity 1:**

Create a GitHub Account. Make a repository with the name “**DSA\_Lab”. Mention the link here after the account creation.**

**https://github.com/MufeezAslam/DSA\_Lab.gitSolution:**

**Activity 2:**

Write any 15 programs that will explain the concepts of pointers.

In this file, you should place the code and its output screenshot.

After completing the activities, Upload the final pdf and code to the “**DSA\_Lab”**repository.

**PROGRAM #01**

**Declaration and intiolization**

#include <iostream>

using namespace std;

int main() {

int var = 5;

// declare pointer variable

int\* pointVar;

// store address of var

pointVar = &var;

// print value of var

cout << "var = " << var << endl;

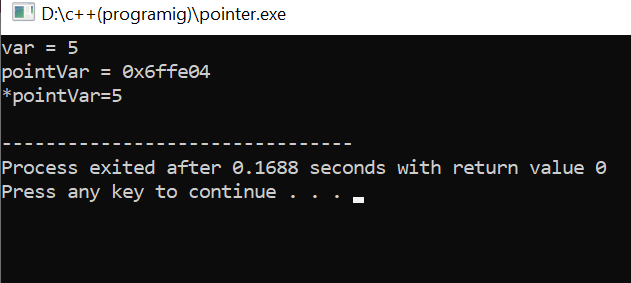
// print pointer pointVar

cout << "pointVar = " << pointVar << endl;

return 0;

**}**

**Output:**



**PROGRAM #02:**

**Assigning value to pointer:**

#include <iostream>

using namespace std;

int main() {

int x = 27;

int \*ip;

ip = &x;

cout << "Value of x is : ";

cout << x << endl;

cout << "Value of ip is : ";

cout << ip<< endl;

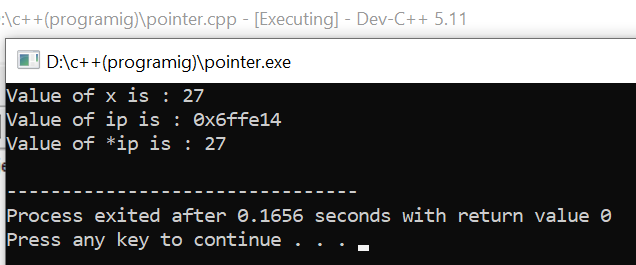
cout << "Value of \*ip is : ";

cout << \*ip << endl;

return 0;

}

**Output:**



**PROGRAM #03:**

### **Changing Value Pointed by Pointers**

#include <iostream>

using namespace std;

int main() {

int var = 5;

int\* pointVar;

// store address of var

pointVar = &var;

// print var

cout << "var = " << var << endl;

// print \*pointVar

cout << "\*pointVar = " << \*pointVar << endl

<< endl;

cout << "Changing value of var to 7:" << endl;

// change value of var to 7

var = 7;

// print var

cout << "var = " << var << endl;

// print \*pointVar

cout << "\*pointVar = " << \*pointVar << endl

<< endl;

cout << "Changing value of \*pointVar to 16:" << endl;

// change value of var to 16

\*pointVar = 16;

// print var

cout << "var = " << var << endl;

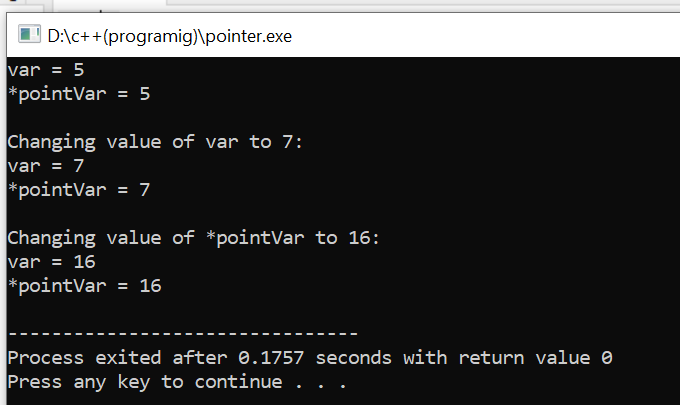
// print \*pointVar

cout << "\*pointVar = " << \*pointVar << endl;

return 0;

}

**Output:**

****

**PROGRAM#04:**

**Pointer and Array:**

#include <iostream>

using namespace std;

int main() {

int arr[] = {1, 2, 3, 4, 5};

int \*ptr = arr;

cout << "Value at ptr: " << \*ptr <<endl;

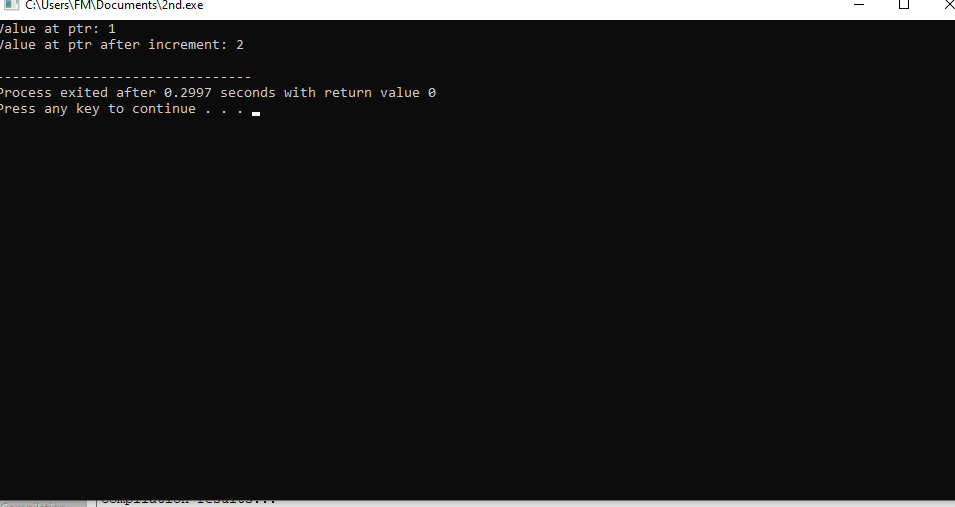
ptr++;

cout << "Value at ptr after increment: " << \*ptr <<endl;

return 0;

}

**Output:**

****

**PROGRAM#05:**

**Addition of pointer:**

#include <iostream>

using namespace std;

int main() {

int num1 = 20;

int num2 = 10;

int \*ptr1 = &num1;

int \*ptr2 = &num2;

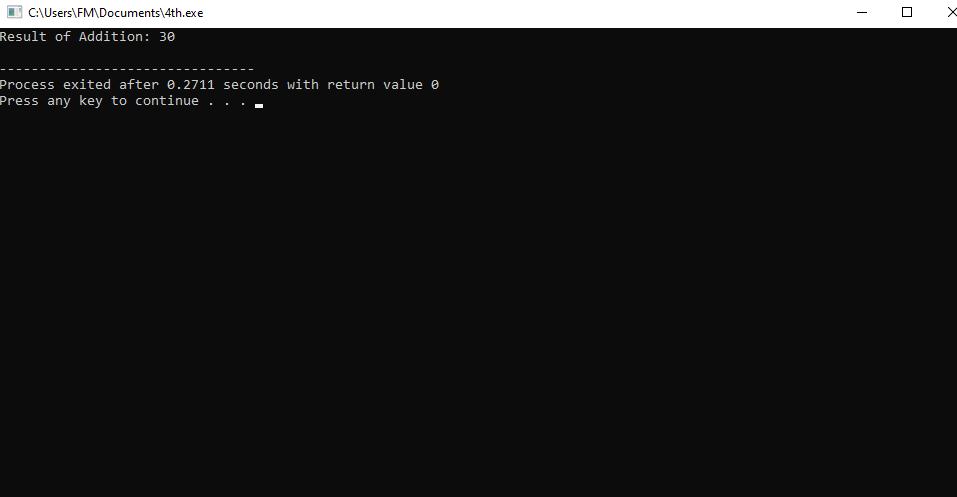
int result = \*ptr1 + \*ptr2;

cout << "Result of Addition: " << result << endl;

return 0;

}

**Output:**

****

**PROGRAM#06:**

**Subtraction of pointer:**

#include <iostream>

using namespace std;

int main() {

int num1 = 20;

int num2 = 10;

int \*ptr1 = &num1;

int \*ptr2 = &num2;

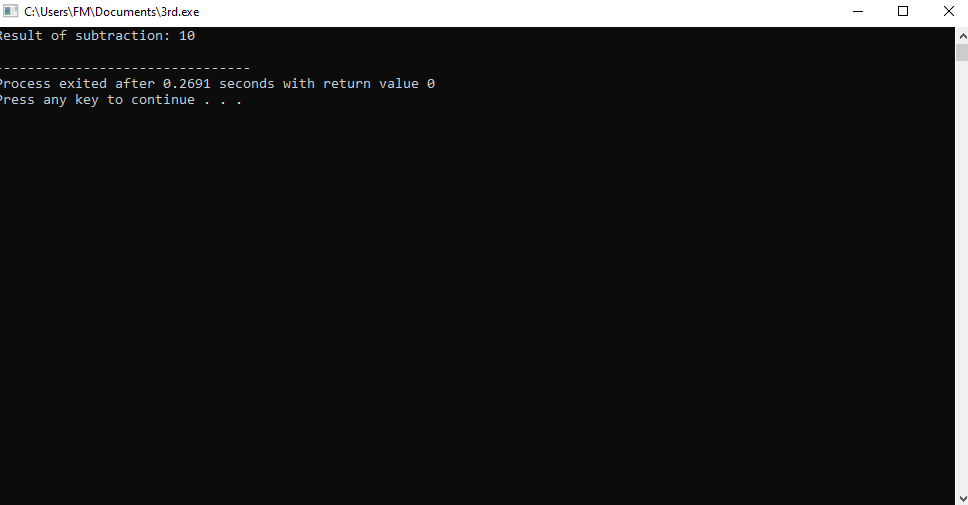
int result = \*ptr1 - \*ptr2;

cout << "Result of subtraction: " << result <<endl;

return 0;

}

**Output:**

****

**PROGRAM#07:**

**Multiplication of pointer :**

#include <iostream>

using namespace std;

int main() {

int num1 = 5;

int num2 = 6;

int \*ptr1 = &num1;

int \*ptr2 = &num2;

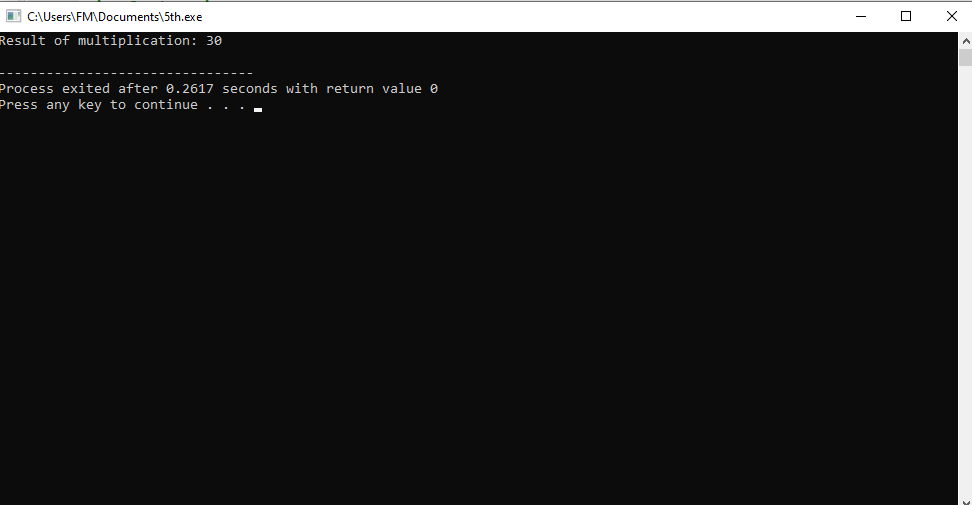
int result = (\*ptr1) \* (\*ptr2);

cout << "Result of multiplication: " << result <<endl;

return 0;

}

**Output:**

****

**PROGRAM#08:**

**Division of pointer:**

#include <iostream>

using namespace std;

int main() {

double num1 = 20.0;

double num2 = 5.0;

double \*ptr1 = &num1;

double \*ptr2 = &num2;

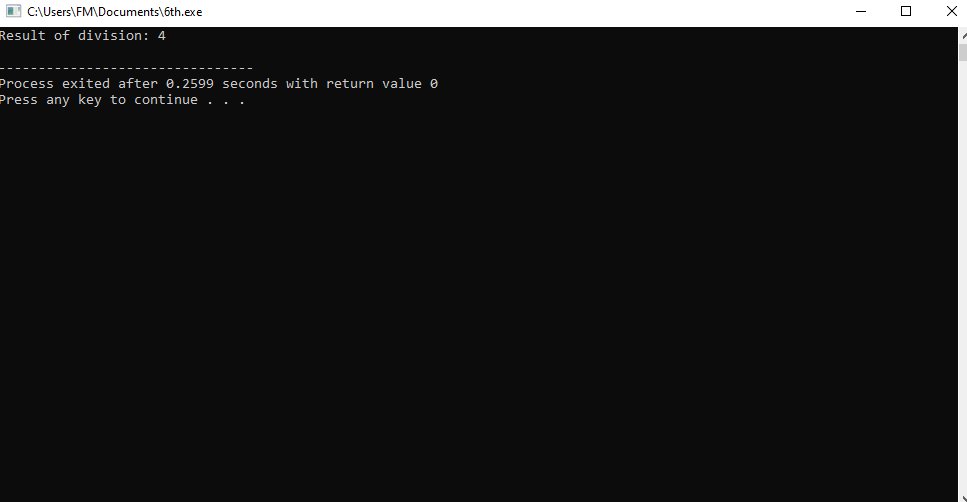
double result = (\*ptr1) / (\*ptr2);

cout << "Result of division: " << result << endl;

return 0;

}

**Output:**

****

**PROGRAM#09:**

**Decrement of pointer:**

#include <iostream>

using namespace std;

int main() {

int num = 10;

int \*ptr = &num;

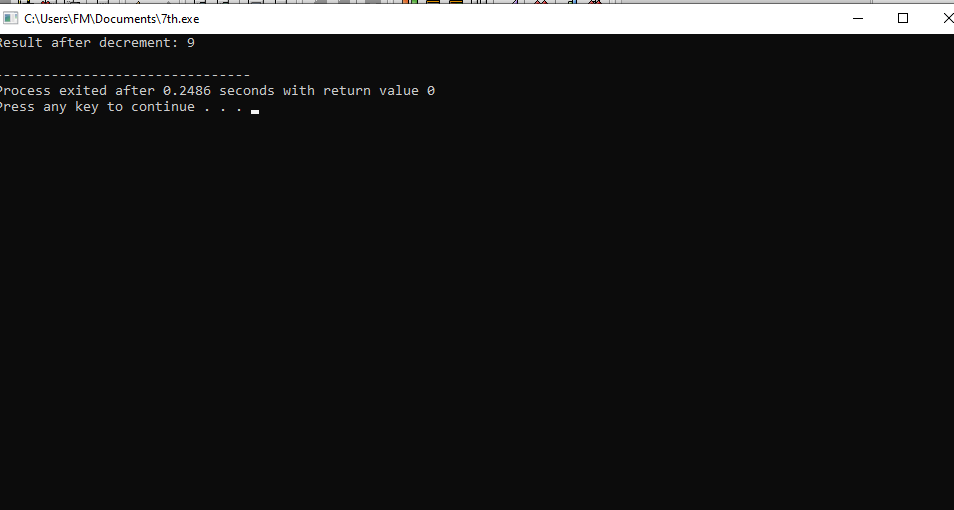
(\*ptr)--;

cout << "Result after decrement: " << \*ptr << endl;

return 0;

}

**Output:**

****

**PROGRAM#10:**

**Pointer to pointer :**

#include <iostream>

using namespace std;

int main() {

int x = 10;

int \*ptr1 = &x;

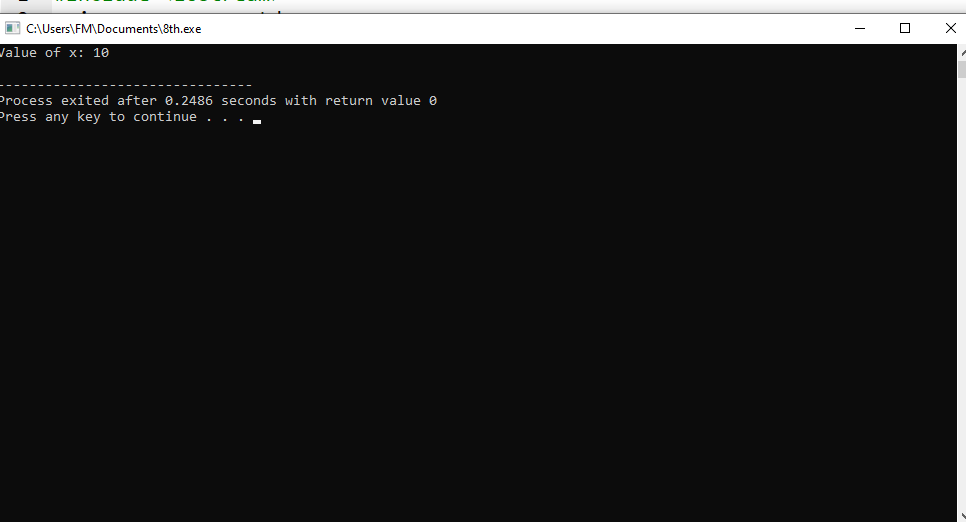
int \*\*ptr2 = &ptr1;

cout << "Value of x: " << \*\*ptr2 << endl;

return 0;

}

**Output:**

****

**PROGRAM#11:**

**Pointer and array(Index element of array):**

#include <iostream>

using namespace std;

int main() {

int arr[] = {1, 2, 3, 4, 5};

int\* ptr = arr;

for (int i = 0; i < 5; i++) {

cout << "Value at index " << i << ": " << \*ptr << endl;

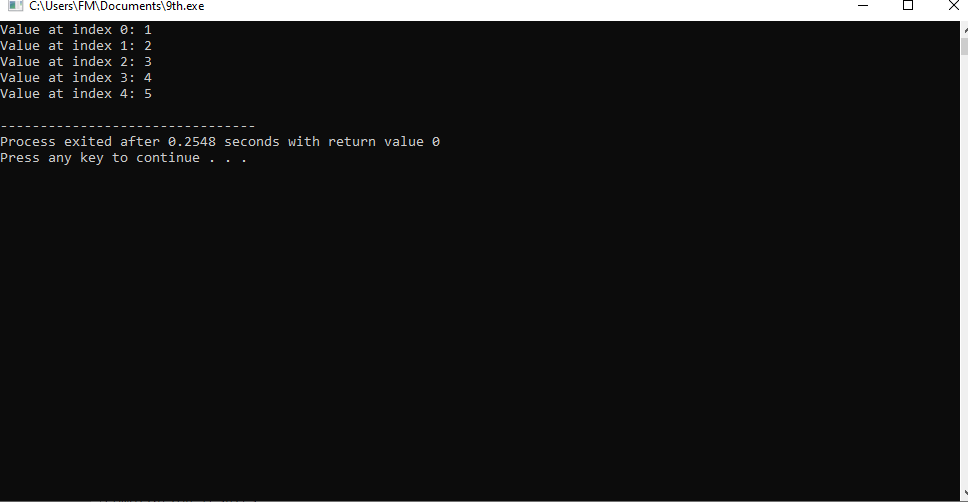
ptr++;

}

return 0;

}

**Output:**

****

**PROGRAM#12:**

**Pointer and function:**

#include <iostream>

using namespace std;

int addNumbers(int\* a, int\* b) {

return \*a + \*b;

}

int main() {

int num1, num2;

cout << "Enter two numbers: ";

cin >> num1 >> num2;

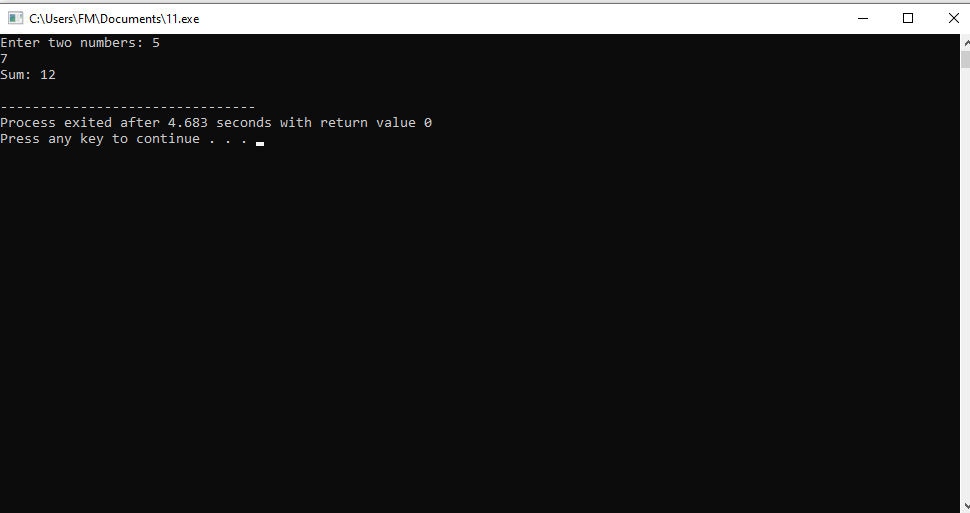
int sum = addNumbers(&num1, &num2);

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

return 0;

}

**Output:**



**PROGRAM#13:**

**Swaping of two pointer:**

#include <iostream>

using namespace std;

void swap(int\* a, int\* b) {

\*a ^= \*b;

\*b ^= \*a;

\*a ^= \*b;

}

int main() {

int num1 = 5;

int num2 = 10;

cout << "Before swapping: num1 ="<<num1<<",num2="<<num2<<endl;

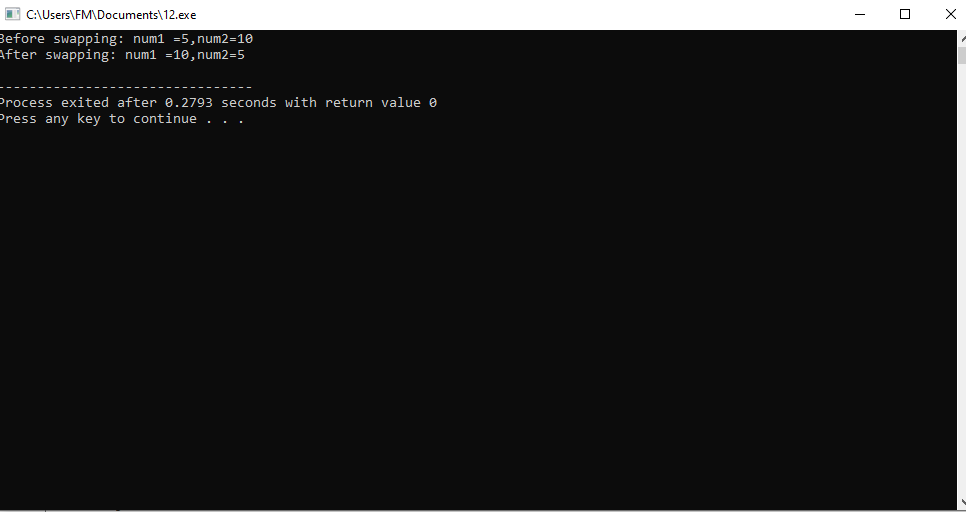
swap(&num1, &num2);

cout << "After swapping: num1 ="<<num1<<",num2="<<num2<<endl;

return 0;

}

**Output:**



**PROGRAM#14:**

**Void pointer:**

#include <iostream>

using namespace std;

int main() {

void\* ptr;

float f = 2.3f;

// assign float address to void

ptr = &f;

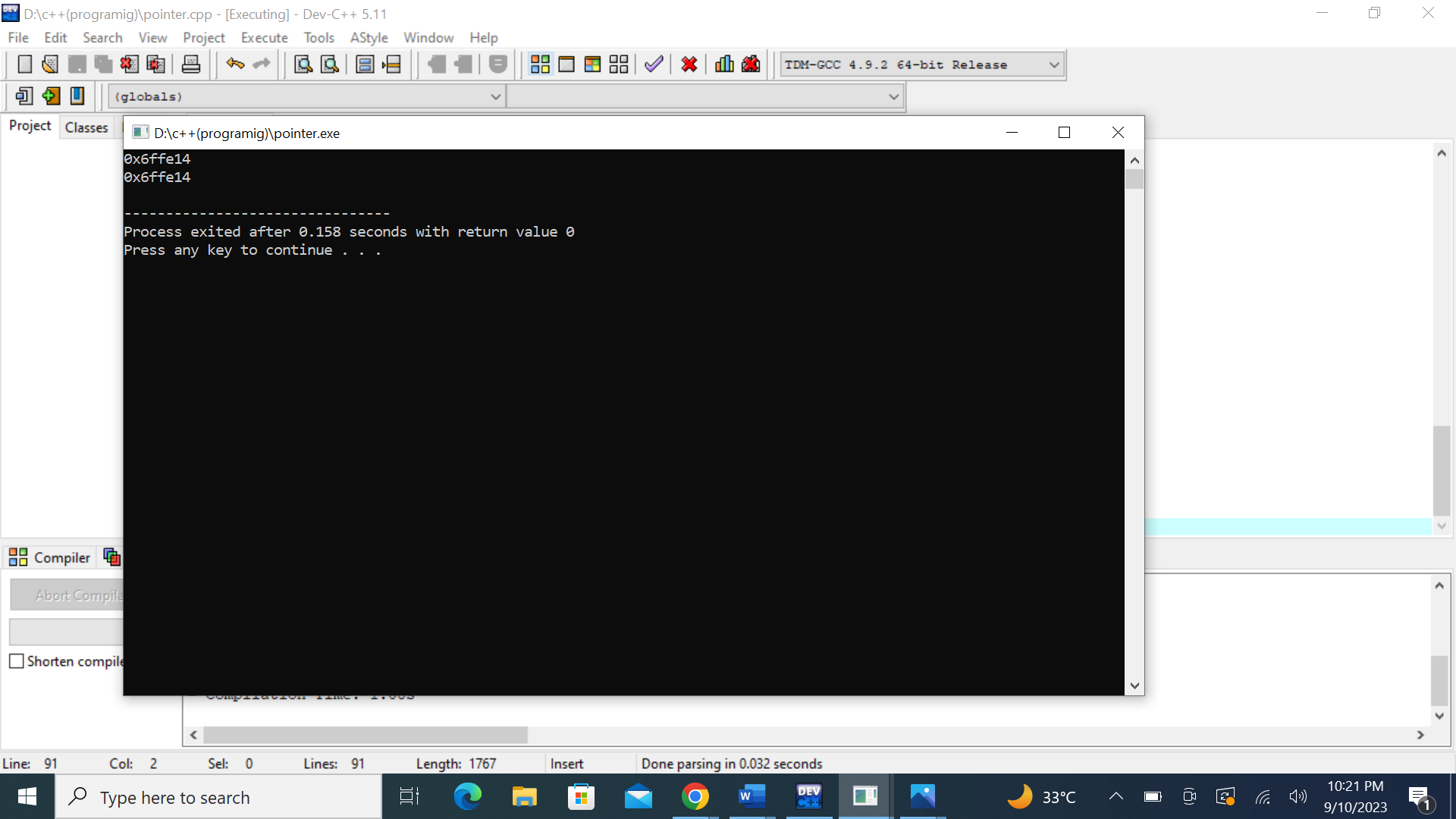
cout << &f << endl;

cout << ptr << endl;

return 0;

}

**Output:**



**PROGRAM#15:**

**Pointer to stracture:**

#include <iostream>

using namespace std;

struct Distance {

int feet;

float inch;

};

int main() {

Distance \*ptr, d;

ptr = &d;

cout << "Enter feet: ";

cin >> (\*ptr).feet;

cout << "Enter inch: ";

cin >> (\*ptr).inch;

cout << "Displaying information." << endl;

cout << "Distance = " << (\*ptr).feet << " feet " << (\*ptr).inch << " inches";

return 0;

}

**Output:**

