

ASSIGNMENT # 1

Program# 1: Addition Program

```
#include <iostream>

using namespace std;

int main() {

    int num1 = 5;

    int num2 = 7;

    int result;

    int* ptr1 = &num1;

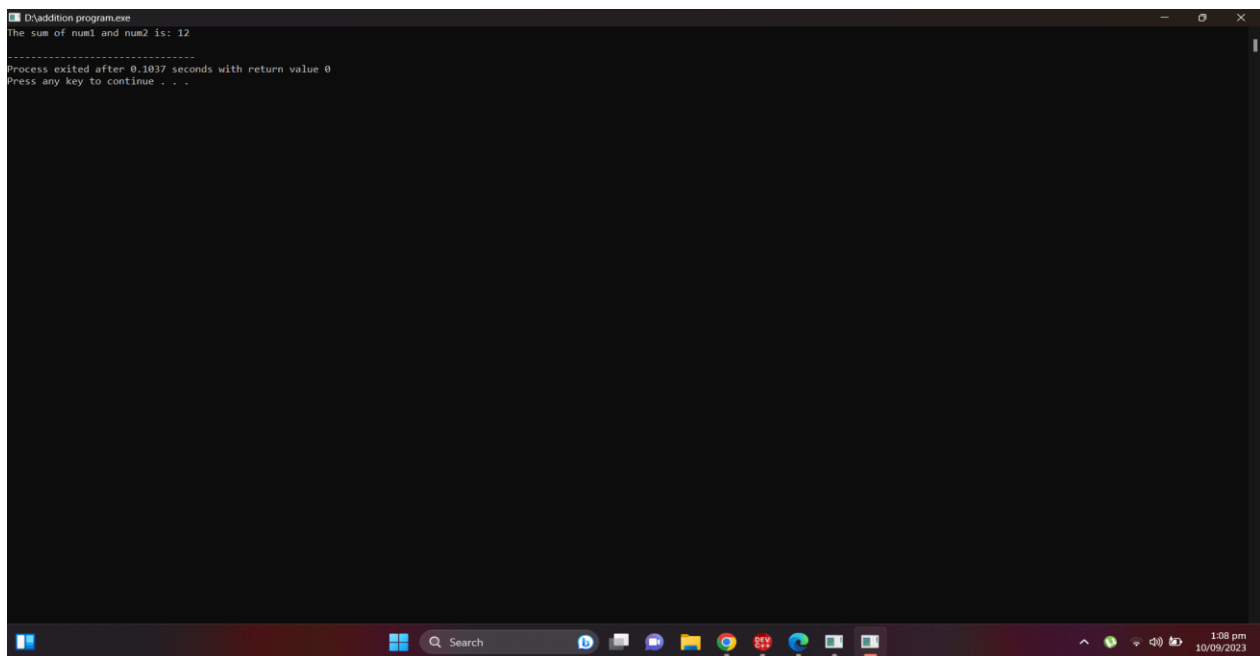
    int* ptr2 = &num2;

    result = (*ptr1) + (*ptr2);

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

    return 0;

}
```

A screenshot of a Windows command prompt window titled "D:\addition program.exe". The window displays the output of the program: "The sum of num1 and num2 is: 12". Below this, it shows "Process exited after 0.1037 seconds with return value 0" and "Press any key to continue . . .". The Windows taskbar is visible at the bottom, showing the Start button, Search bar, and several application icons. The system clock in the bottom right corner indicates "1:08 pm 10/09/2023".

Program# 2: Subtraction Program

```
#include <iostream>

using namespace std;

int main () {

    int num1 = 10;

    int num2 = 4;

    int result;

    int* ptr1 = &num1;

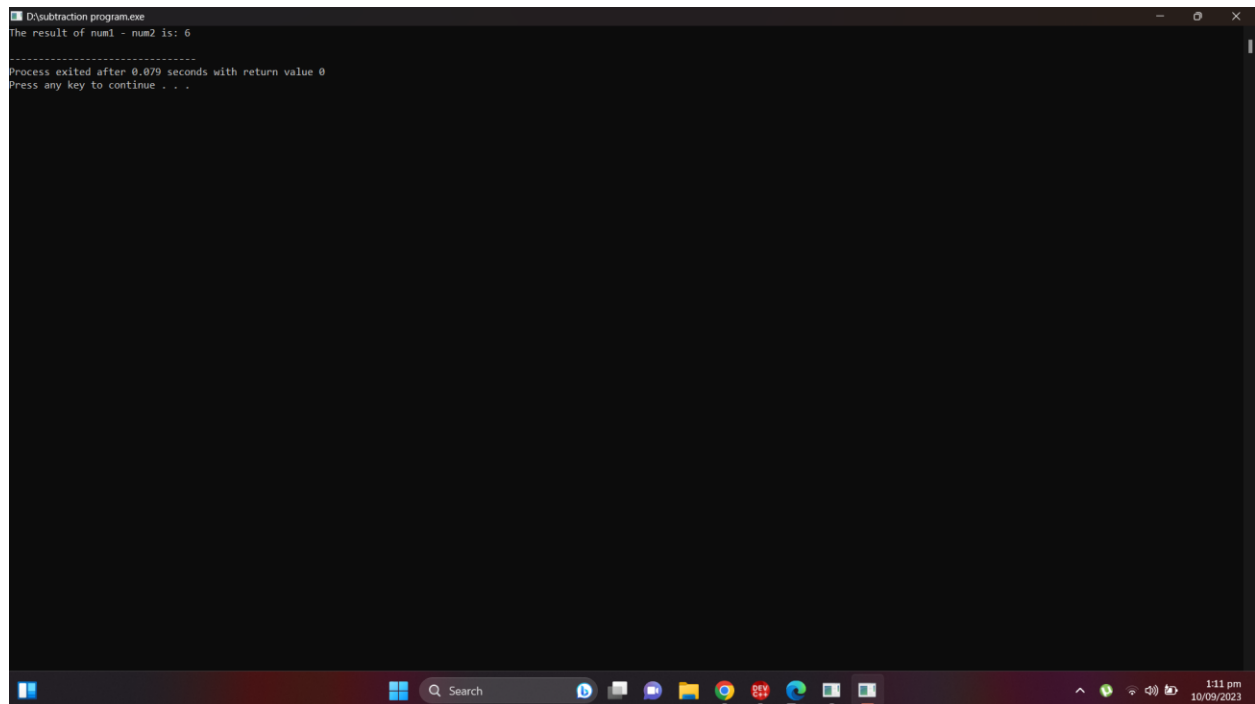
    int* ptr2 = &num2;

    result = (*ptr1) - (*ptr2);

    cout << "The result of num1 - num2 is: " << result << endl;

    return 0;

}
```



The screenshot shows a Windows command prompt window titled "D:\subtraction program.exe". The output of the program is displayed as follows:

```
-----
The result of num1 - num2 is: 6
-----
Process exited after 0.079 seconds with return value 0
Press any key to continue . . .
```

The window is running on a Windows 10 desktop, with the taskbar visible at the bottom showing various application icons and the system clock indicating 1:11 pm on 10/09/2023.

Program# 3: Increment Program

```
#include <iostream>

using namespace std;

int main () {

    int num = 5;

    int* ptr = &num;

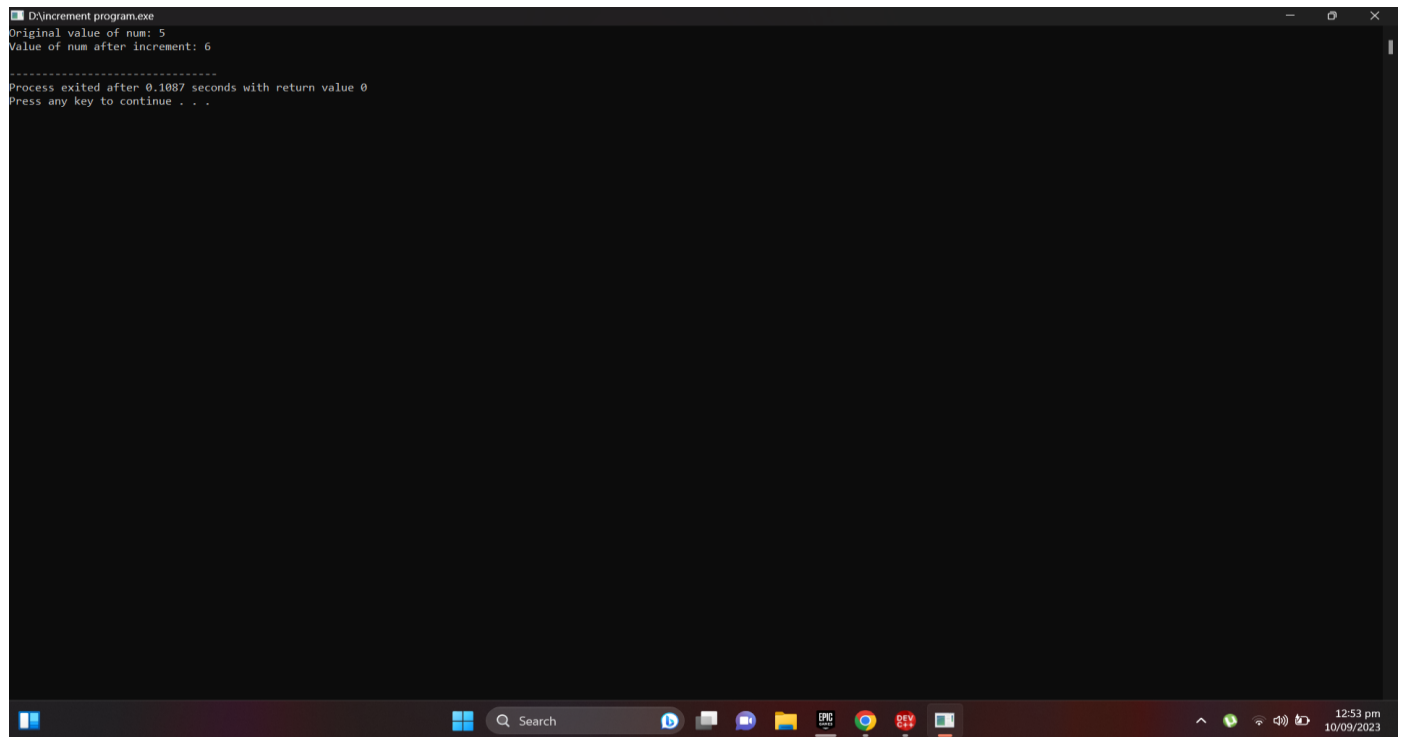
    cout << "Original value of num: " << num << endl;

    (*ptr)++;

    cout << "Value of num after increment: " << num << endl;

    return 0;

}
```



The screenshot shows a Windows command prompt window titled "D:\increment program.exe". The output of the program is displayed as follows:

```
D:\increment program.exe
Original value of num: 5
Value of num after increment: 6
.....
Process exited after 0.1087 seconds with return value 0
Press any key to continue . . .
```

The window is running on a Windows 10 desktop, with the taskbar visible at the bottom showing the Start button, Search bar, and several application icons including Edge, File Explorer, and Chrome. The system clock in the bottom right corner indicates the time is 12:53 pm on 10/09/2023.

Program# 4: Decrement Program

```
#include <iostream>

using namespace std;

int main () {

    int num = 10;

    int* ptr = &num;

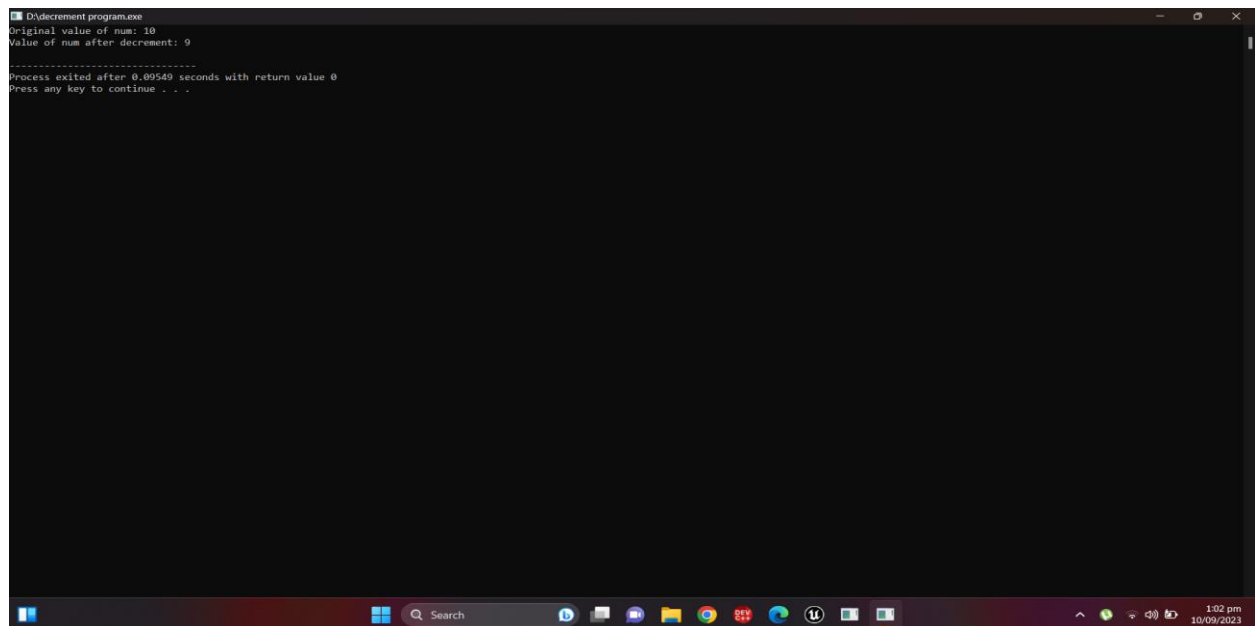
    cout << "Original value of num: " << num << endl;

    (*ptr)--;

    cout << "Value of num after decrement: " << num << endl;

    return 0;

}
```



```
D:\decrement program.exe
Original value of num: 10
Value of num after decrement: 9

-----
Process exited after 0.09549 seconds with return value 0
Press any key to continue . . .
```

Program# 5: Comparison Program

```
#include <iostream>

using namespace std;

int main() {

    int num1 = 10;

    int num2 = 20;

    int* ptr1 = &num1;

    int* ptr2 = &num2;

    if (ptr1 == ptr2) {

        cout << "ptr1 and ptr2 point to the same location in memory." <<endl;

    } else {

        cout << "ptr1 and ptr2 do not point to the same location in memory." <<endl;

    }

    if (ptr1 > ptr2) {

        cout << "ptr1 is greater (points to a higher memory address) than ptr2." <<endl;

    } else if (ptr1 < ptr2) {

        cout << "ptr1 is smaller (points to a lower memory address) than ptr2." <<endl;

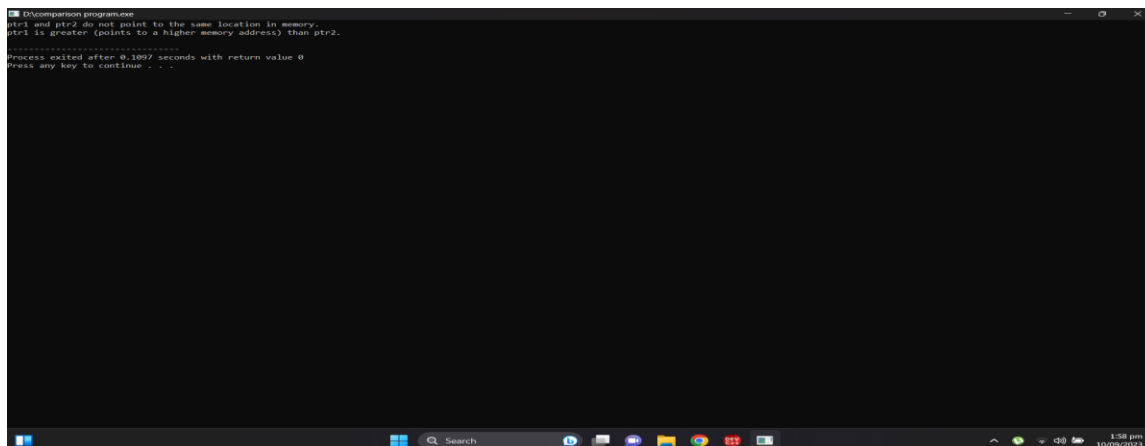
    } else {

        cout << "ptr1 and ptr2 have the same memory address." <<endl;

    }

    return 0;

}
```



```
D:\comparison program.exe
ptr1 and ptr2 do not point to the same location in memory.
ptr1 is greater (points to a higher memory address) than ptr2.
Press any key to continue . . .
Process exited after 0.1097 seconds with return value 0
Press any key to continue . . .
```

Program# 6: Arithmetic with structs program Program

```
#include <iostream>

#include <string>

using namespace std;

struct Person {
    string name;

    int age;
};

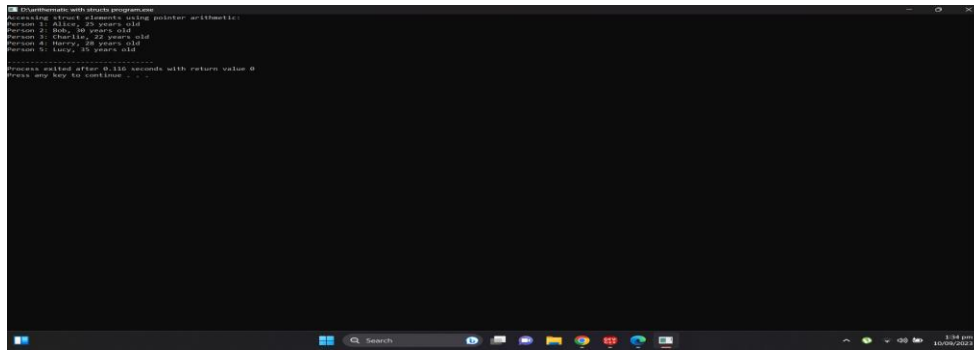
int main () {
    Person people [] = {
        {"Alice", 25},
        {"Bob", 30},
        {"Charlie", 22},
        {"Harry", 28},
        {"Lucy", 35}
    };

    Person* ptr = people;

    cout << "Accessing struct elements using pointer arithmetic:" << endl;

    for (int i = 0; i < 5; i++)
    {
        cout << "Person " << i + 1 << ": " << (ptr + i)->name << ", " << (ptr + i)->age << " years old" << endl;
    }

    return 0;
}
```



Program# 7: Arithmetic with Class program

```
#include <iostream>

#include <string>

using namespace std;

class Person {
public:
    Person (const string& name, int age) : name(name), age(age) {}

    void printInfo () {
        cout << "Name: " << name << ", Age: " << age << " years old" << endl;}
private:
    string name;
    int age;
};

int main () {
    Person people [] = {
        {"Alice", 25},
        {"Bob", 30},
        {"Charlie", 22},
        {"David", 28},
        {"Eve", 35}
    };

    Person* ptr = people;
```

```

cout<< "Accessing class object elements using pointer arithmetic:" <<endl;

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

    cout << "Person " << i + 1 << ": ";

    (ptr + i)->printInfo();

}

return 0;

}

```

```

D:\Arithmetic with class program.exe
Accessing class object elements using pointer arithmetic:
Person 1: Name: Alice, Age: 25 years old
Person 2: Name: Bob, Age: 30 years old
Person 3: Name: Charlie, Age: 22 years old
Person 4: Name: David, Age: 28 years old
Person 5: Name: Eve, Age: 35 years old
-----
Process exited after 0.00704 seconds with return value 0
Press any key to continue . . .

```

Program# 8: Adding 5 numbers using pointers program

```

#include <iostream>

using namespace std;

int main () {

    int numbers [5];

    cout << "Enter 5 numbers:" <<endl;

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

        cout<< "Enter number " << i + 1 << ": ";

        cin >> numbers[i];

    }

    int* ptr = numbers;

    int sum = 0;

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

```



```

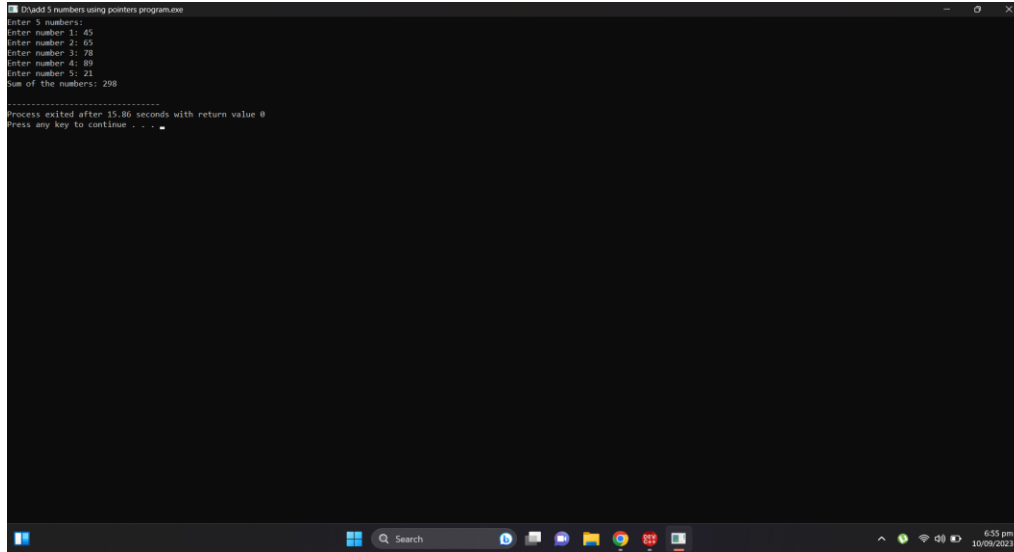
        sum += *ptr;

        ptr++;
    }

    cout << "Sum of the numbers: " << sum << endl;

    return 0;
}

```



```

D:\add 5 numbers using pointers program.exe
Enter 5 numbers:
Enter number 1: 45
Enter number 2: 65
Enter number 3: 78
Enter number 4: 89
Enter number 5: 21
Sum of the numbers: 298

-----
Process exited after 15.86 seconds with return value 0
Press any key to continue . . .

```

Program# 9: Arithmetic for dynamic memory allocation program

```

#include <iostream>

using namespace std;

int main() {

    int size;

    cout << "Enter the size of the array: ";

    cin >> size;

    int* dynamicArray = new int[size];

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

        dynamicArray[i] = i * 10;

    }

    cout << "Array contents using pointer arithmetic:" << endl;

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

```

```

        cout << *(dynamicArray + i) << " ";
    }
    cout << endl;
    delete[] dynamicArray;
    return 0;
}

```

```

D:\arithmetic for dynamic memory allocation program.exe
Enter the size of the array: 10
Array contents using pointer arithmetic:
0 10 20 30 40 50 60 70 80 90
Press any key to continue . . .
Process exited after 2.862 seconds with return value 0

```

Program# 10: Arithmetic for byte-level manipulation program

```

#include <iostream>

using namespace std;

int main() {
    char buffer[5] = {0x41, 0x42, 0x43, 0x44, 0x45};
    char* ptr = buffer;

    cout << "Original buffer contents: ";
    for (int i = 0; i < 5; i++) {
        cout << static_cast<int>(buffer[i]) << " ";
    }
    cout << endl;

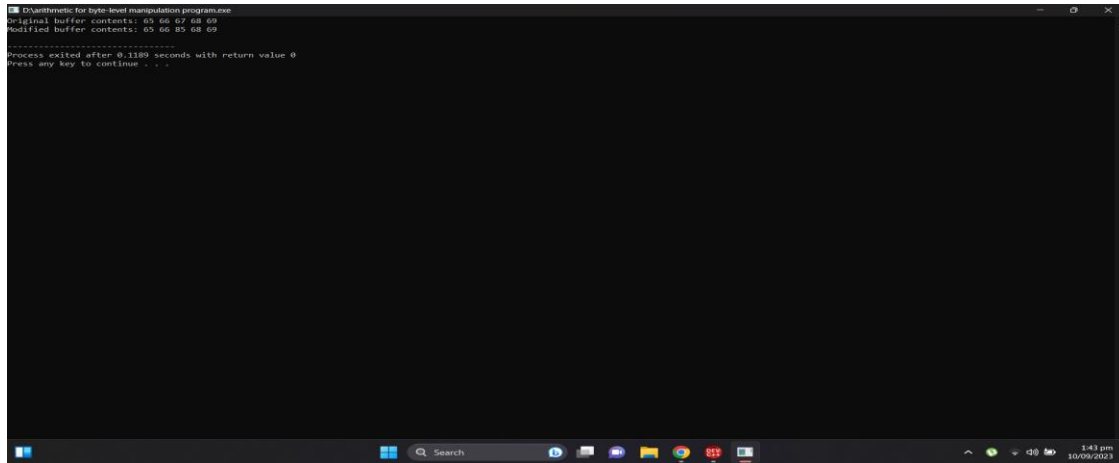
    *(ptr + 2) = 0x55;
    cout << "Modified buffer contents: ";
}

```

```

for (int i = 0; i < 5; i++) {
    cout << static_cast<int>(buffer[i]) << " ";
}
cout << endl;
return 0;
}

```



```

D:\Arithmetic for byte-level manipulation program.exe
Original buffer contents: 65 66 67 68 69
Modified buffer contents: 85 86 87 88 89
Process exited after 0.1189 seconds with return value 0
Press any key to continue . . .

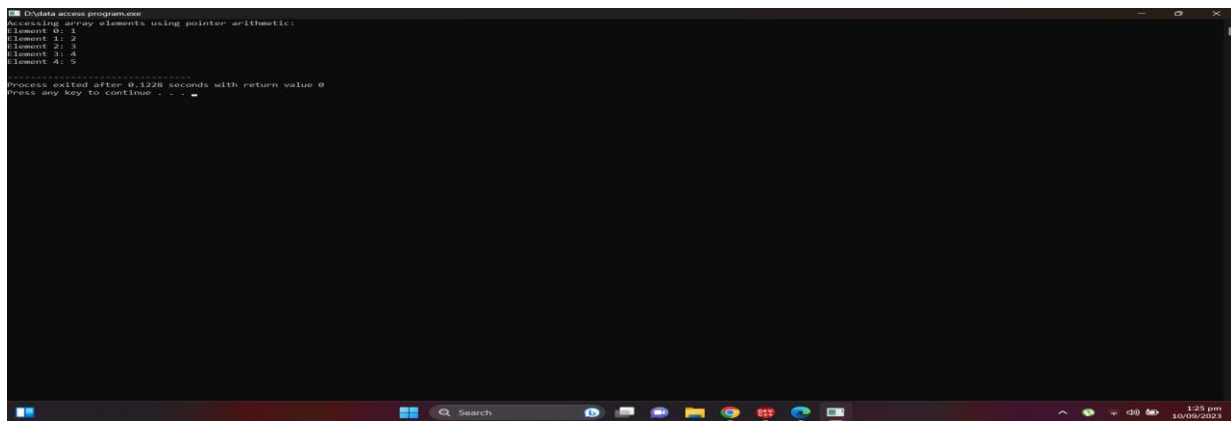
```

Program# 11: Data Access Program

```

#include <iostream>
using namespace std;
int main() {
    int numbers[] = {1, 2, 3, 4, 5};
    int* ptr = numbers;
    cout<<"Accessing array elements using pointer arithmetic:" <<endl;
    for (int i = 0; i < 5; i++) {
        cout<< "Element " << i << ": " << *(ptr + i) <<endl;
    }
    return 0;
}

```



Program# 12: Pointers to swap two numbers program

```
#include <iostream>

using namespace std;

void swapNumbers(int* a, int* b) {

    int temp = *a;

    *a = *b;

    *b = temp;

}

int main() {

    int num1 = 10;

    int num2 = 20;

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

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

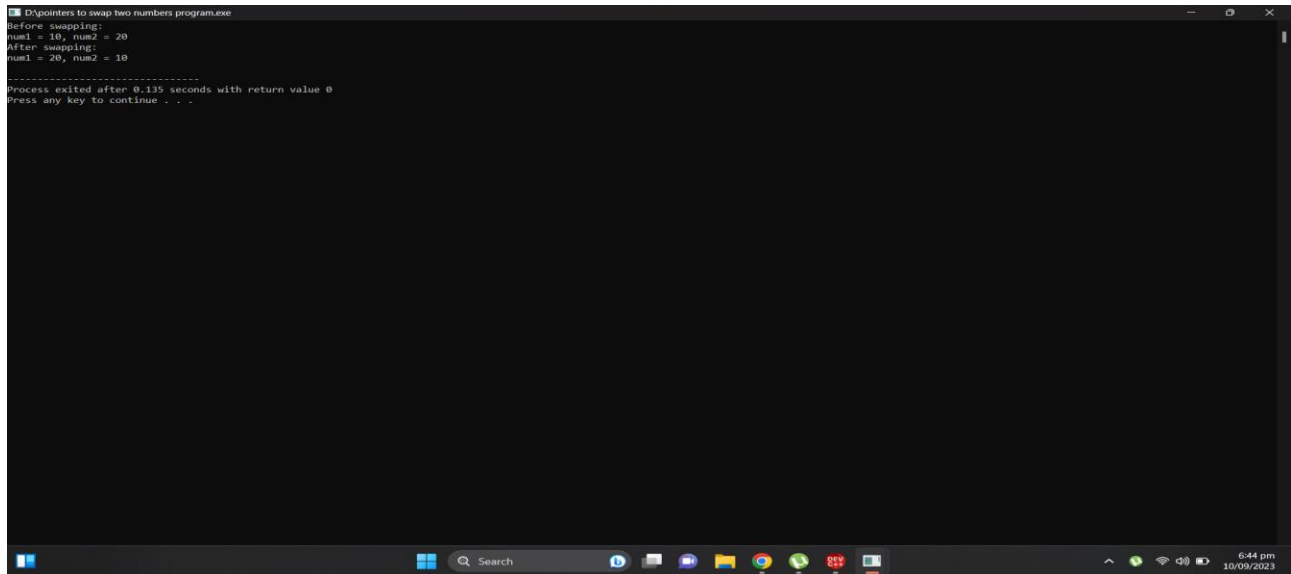
    swapNumbers(&num1, &num2);

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

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

    return 0;

}
```



Program# 13: Increments 5 numbers using pointers program

```
#include <iostream>

using namespace std;

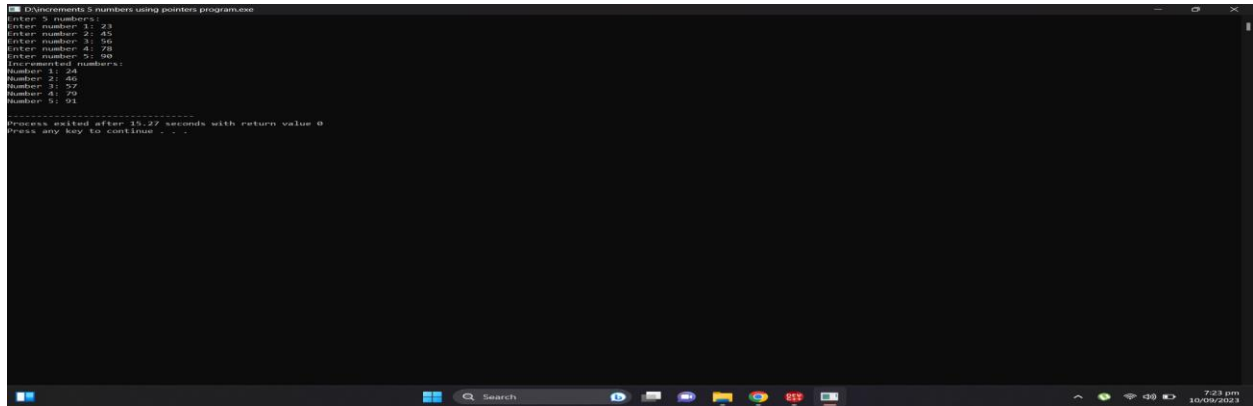
int main() {
    int numbers[5];

    cout << "Enter 5 numbers:" << endl;
    for (int i = 0; i < 5; ++i) {
        cout << "Enter number " << i + 1 << ": ";
        cin >> numbers[i];
    }

    int* ptr = numbers;
    for (int i = 0; i < 5; ++i) {
        (*ptr)++;
        ptr++;
    }

    cout << "Incremented numbers:" << endl;
    for (int i = 0; i < 5; ++i) {
        cout << "Number " << i + 1 << ": " << numbers[i] << endl;
    }
}
```

```
    return 0;
}
```



```
Incrementing 5 numbers using pointers program.exe
Enter 5 numbers:
Enter number 1: 23
Enter number 2: 45
Enter number 3: 56
Enter number 4: 78
Enter number 5: 99
Incremented numbers:
Number 1: 24
Number 2: 46
Number 3: 57
Number 4: 79
Number 5: 99
Process ended after 15.27 seconds with return value 0
Press any key to continue . . .
```

Program# 14: Pointers to manipulate integer values program

```
#include <iostream>
using namespace std;

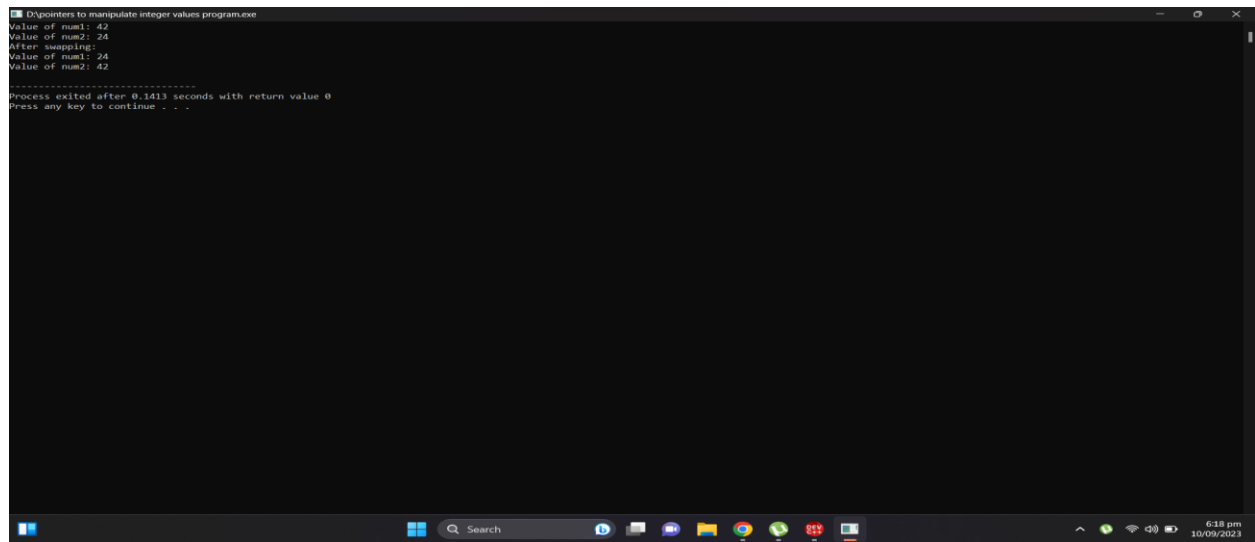
int main() {
    int num1 = 42;
    int num2 = 24;
    int* ptr1 = &num1;
    int* ptr2 = &num2;

    cout << "Value of num1: " << *ptr1 << endl;
    cout << "Value of num2: " << *ptr2 << endl;

    int temp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = temp;

    cout << "After swapping:" << endl;
    cout << "Value of num1: " << *ptr1 << endl;
    cout << "Value of num2: " << *ptr2 << endl;

    return 0;
}
```



```
D:\pointers to manipulate integer values program.exe
Value of num1: 42
Value of num2: 24
After swapping:
Value of num1: 24
Value of num2: 42
-----
Process exited after 0.1413 seconds with return value 0
Press any key to continue . . .
```

Program# 15: Pointers to manipulate variables 2nd program

```
#include <iostream>

using namespace std;

int main() {

    int number = 42;

    int* pointer = &number;

    cout << "Value of number: " << *pointer << endl;

    *pointer = 100;

    cout << "Updated value of number: " << number << endl;

    return 0;

}
```

```
D:\pointers to manipulate variables 2nd program.exe
Value of number: 42
Updated value of number: 100
-----
Process exited after 0.1336 seconds with return value 0
Press any key to continue . . .
```