ASSIGNMENT # 01

Submitted to : Mam Yasmeen Jana

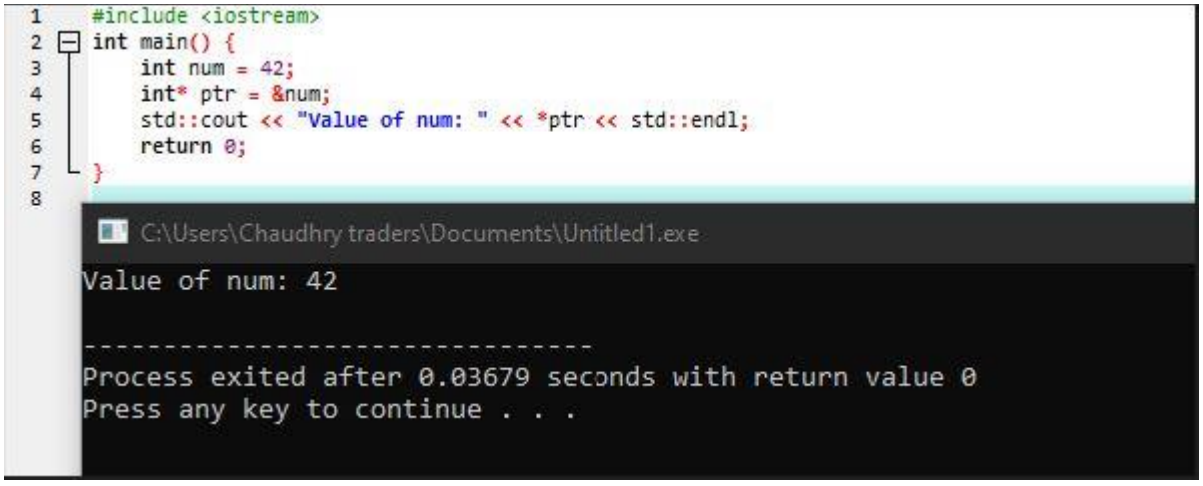
Submitted by : **M.Subhan Nagra**

Roll No : FA21-BCS-054

Subject : DSA-LAB

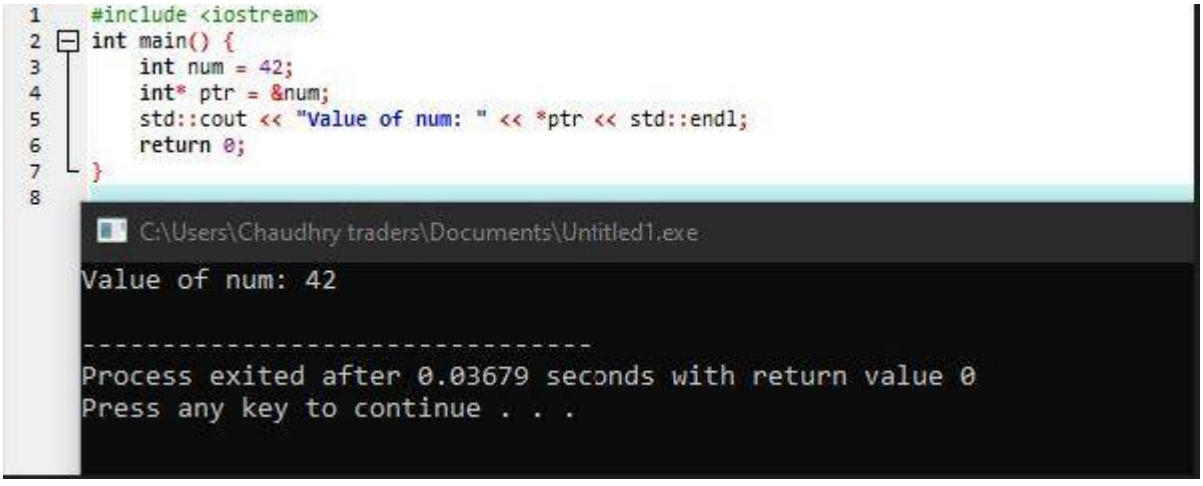
Program 01:

```
#include <iostream>

int main () {

   int num = 42;

   int* ptr = &num;

   std::cout << "Value of num: " << *ptr << std::endl;

   return 0;

}
```



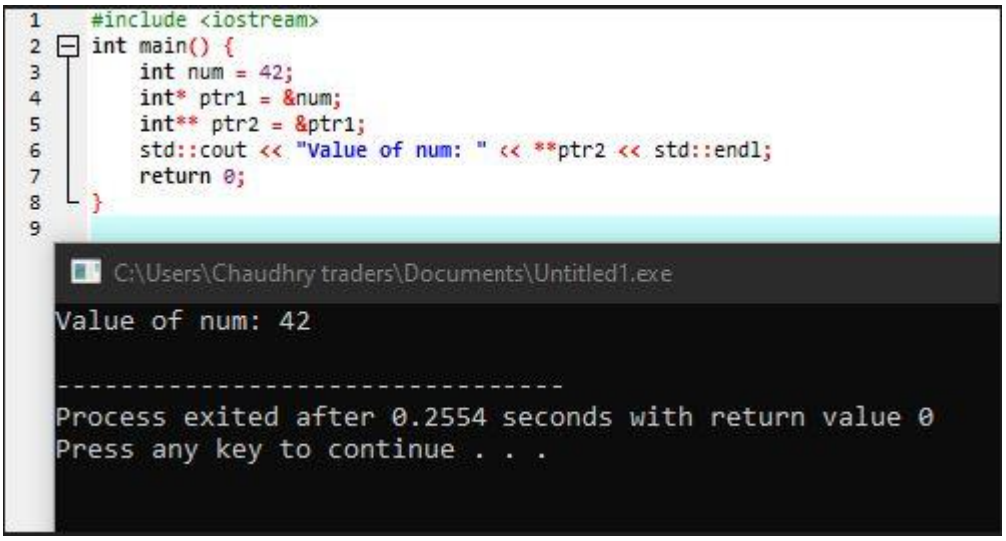
Program 02:

```
#include <iostream>

int main () {

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

   int* ptr = arr;

   std::cout << "First element: " << *ptr << std:: endl;

   std::cout << "Second element: " << *(ptr + 1) <<
std::endl ;

   return 0;

}
```



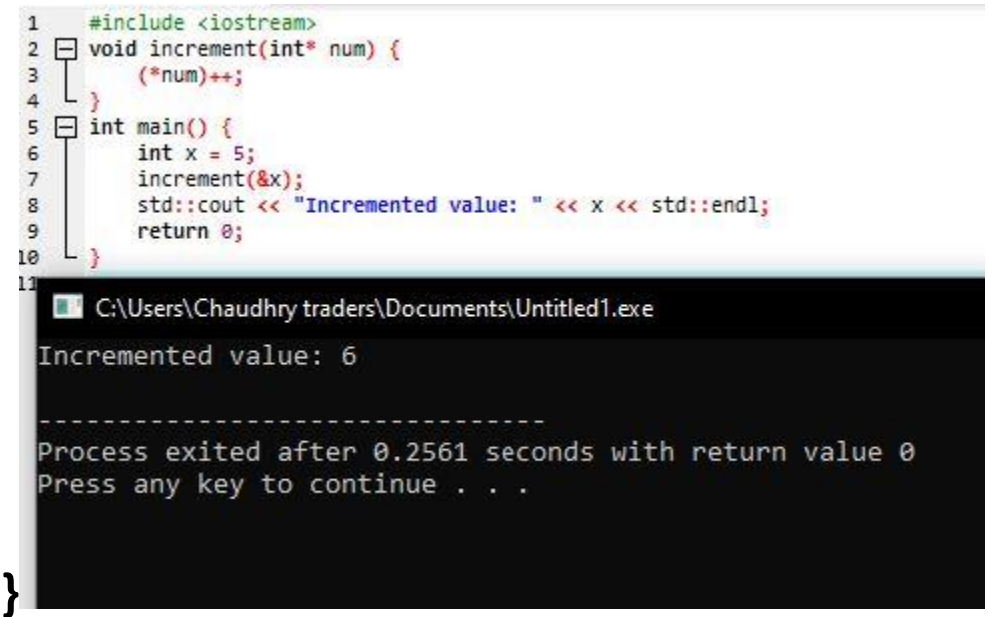
Program 03:

```
#include <iostream>

int main () {

    int num = 42;

    int* ptr1 = &num;

    int** ptr2 = &ptr1;

    std::cout << "Value of num: " << **ptr2 << std::endl;

    return 0;

}
```



Program 04:

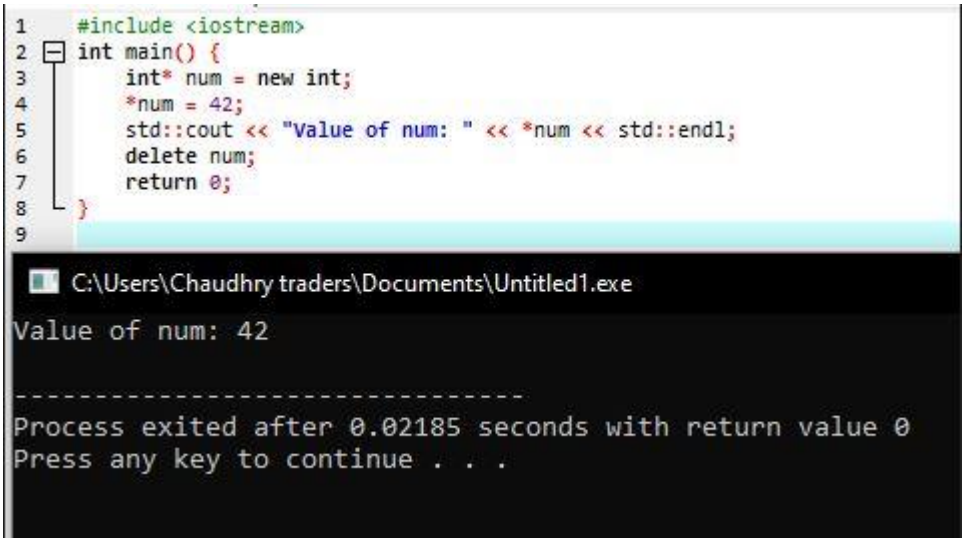
```
#include <iostream>

void increment (int* num) {

   (*num)++;

}

int main() {

   int x = 5;

   increment(&x);

   std::cout << "Incremented value: " << x << std::endl;

   return 0;
```




```
}
```

Program 05:

```
#include <iostream>
int main () {
   int* num = new int;
   *num = 42;
   std::cout << "Value of num: " << *num << std::endl;
   delete num;
   return 0;
}
```



Program 06:

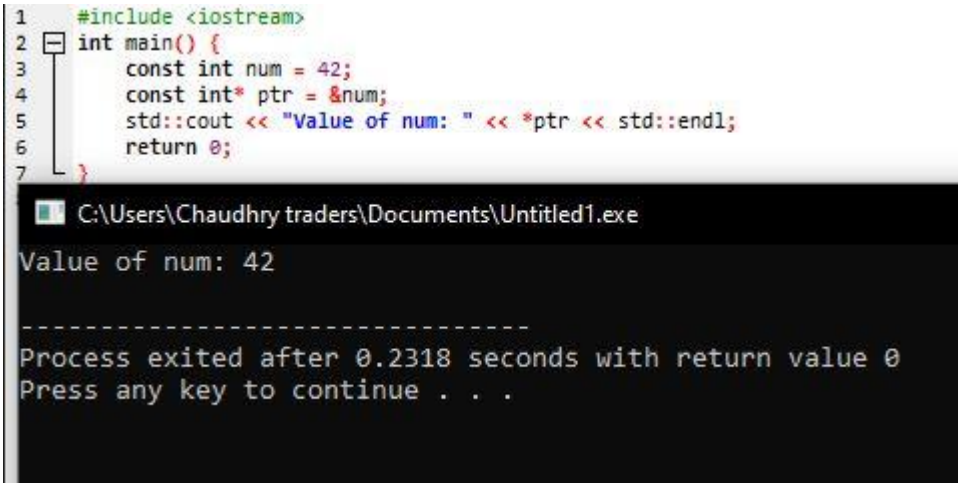
```
#include <iostream>
int main () {
   int arr[] = {1, 2, 3};
   int* ptr[3];
   for (int i = 0; i < 3; i++) {
      ptr[i] = &arr[i];
      std::cout << "Value at index " << i << ": " << *ptr[i]
<< std::endl;
   }
   return 0;
}
```

```
#include <iostream>
int main() {
    int arr[] = {1, 2, 3};
    int* ptr[3];
    for (int i = 0; i < 3; i++) {
        ptr[i] = &arr[i];
        std::cout << "Value at index " << i << ": " << *ptr[i] << std::endl;
    }
    return 0;
}
```

```
C:\Users\Chaudhry traders\Documents\Untitled1.exe
Value at index 0: 1
Value at index 1: 2
Value at index 2: 3

--------------------------------
Process exited after 0.2347 seconds with return value 0
Press any key to continue . . .
```

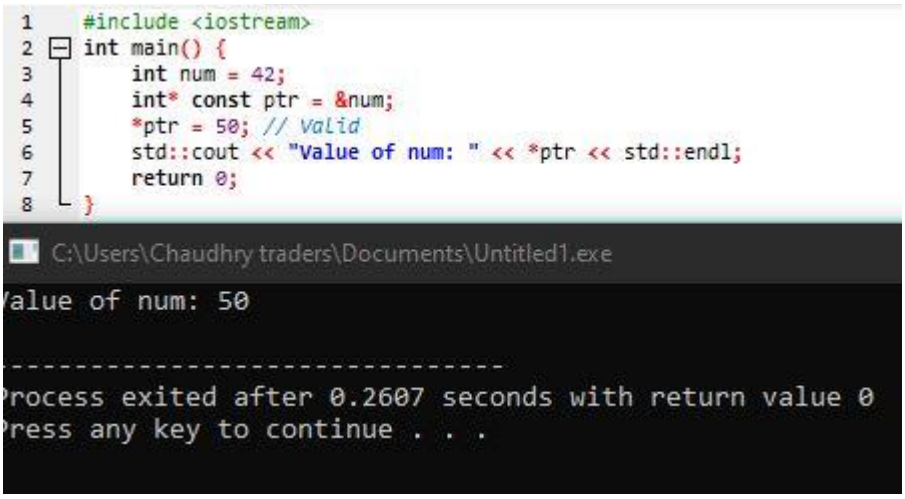
Program 07:

```
#include <iostream>
int main () {
    const int num = 42;
    const int* ptr = &num;
    std::cout << "Value of num: " << *ptr << std::endl;
    return 0;
}
```



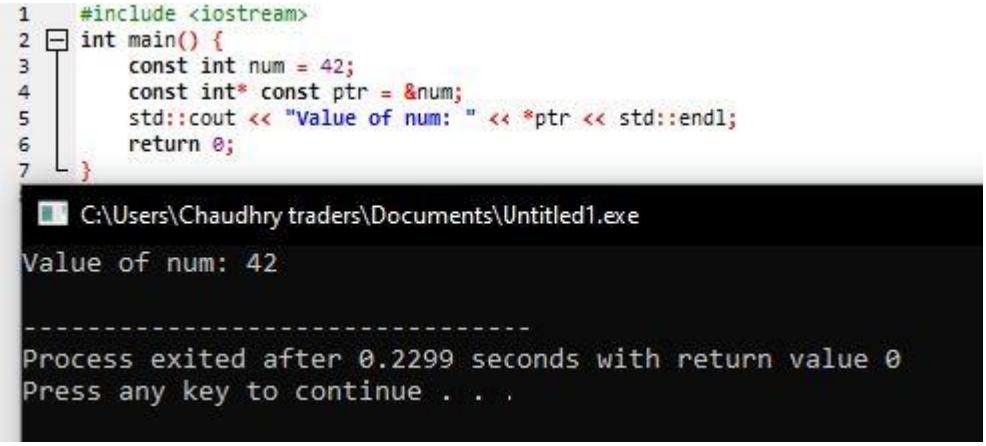
Program 08:

```
#include <iostream>
int main () {
    int num = 42;
    int* const ptr = &num;
    *ptr = 50; // Valid
    std::cout << "Value of num: " << *ptr << std::endl;
    return 0;
}
```



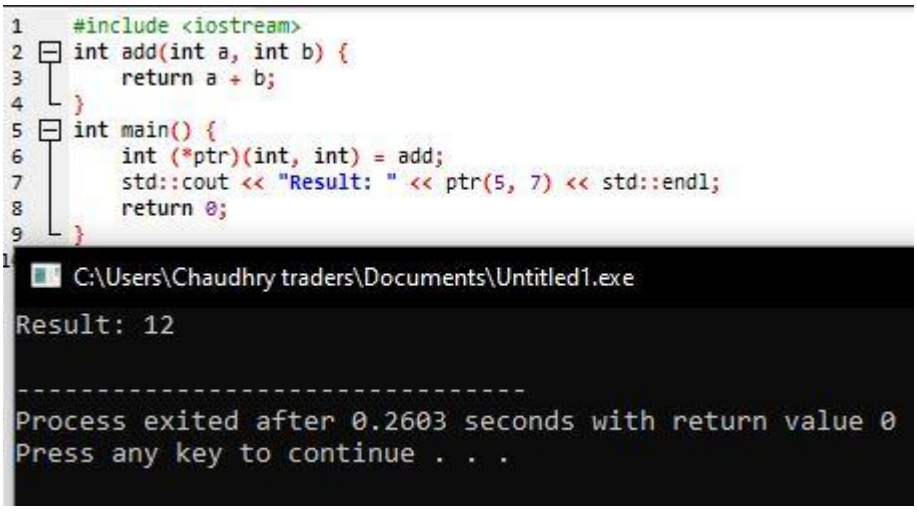
Program 09:

```
#include <iostream>
int main () {
   const int num = 42;
   const int* const ptr = &num;
   std::cout << "Value of num: " << *ptr << std::endl;
   return 0;
}
```



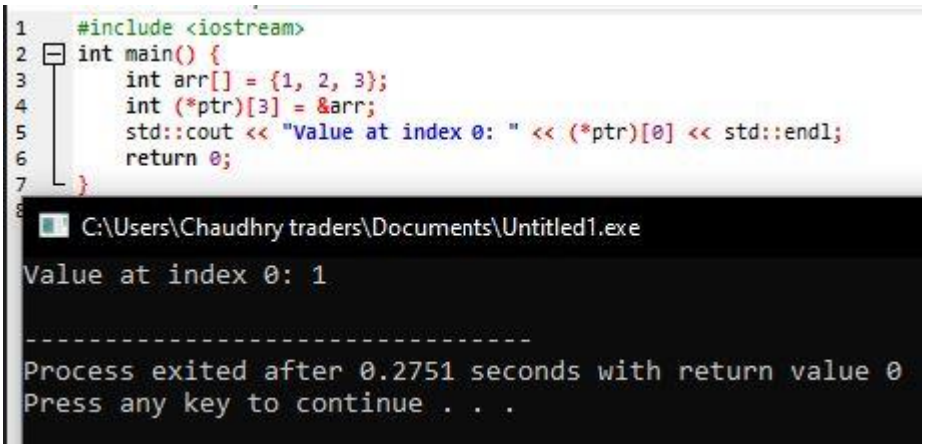
Program 10:

```
#include <iostream>
int add (int a, int b) {
    return a + b;
}
int main() {
    int (*ptr)(int, int) = add;
    std::cout << "Result: " << ptr(5, 7) << std::endl;
    return 0;
}
```



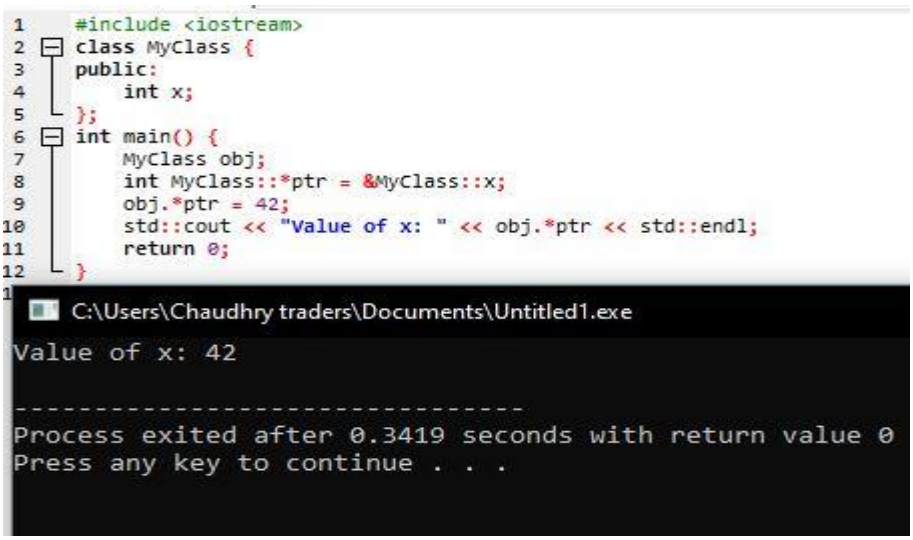
Program 11:

```
#include <iostream>

int main () {

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

   int (*ptr)[3] = &arr;

   std::cout << "Value at index 0: " << (*ptr)[0] << std::endl;

   return 0;

}
```



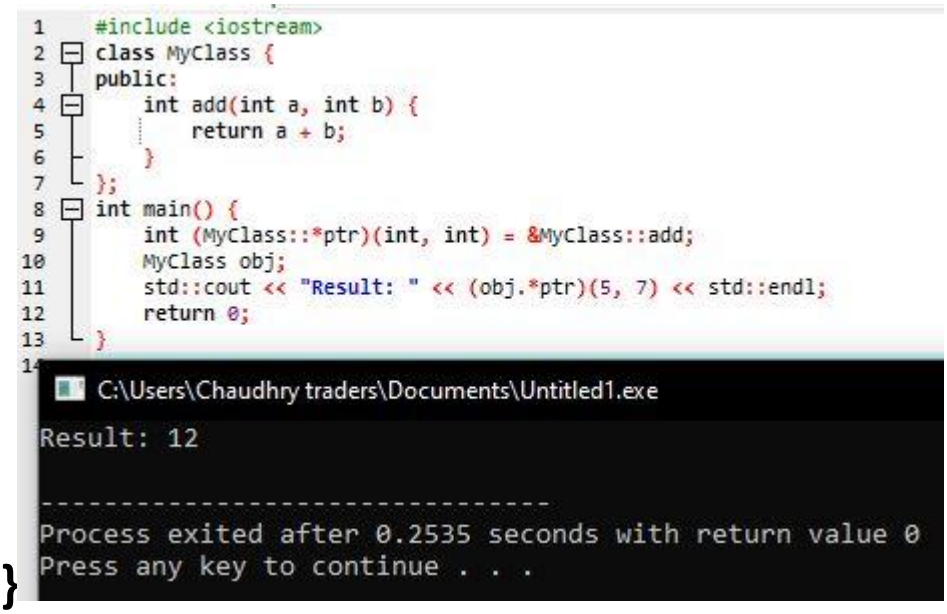
Program 12:

```
#include <iostream>
class MyClass {
public:
    int x;
};
int main() {
    MyClass obj;
    int MyClass::*ptr = &MyClass::x;
    obj.*ptr = 42;
    std::cout << "Value of x: " << obj.*ptr << std::endl;
return 0; }
```



Program 13:

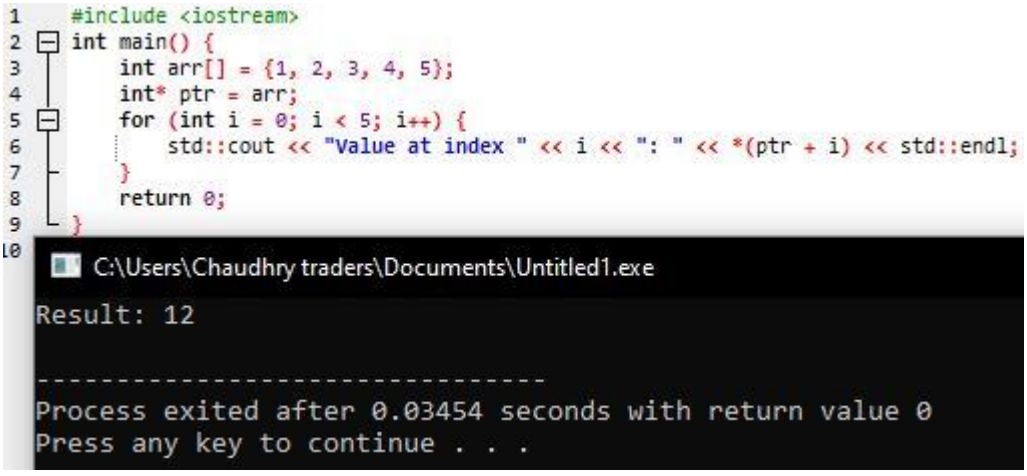
```
#include <iostream>
class MyClass {
public:
    int add(int a, int b) {
        return a + b; }
    };
int main() {
    int (MyClass::*ptr)(int, int) = &MyClass::add;
    MyClass obj;
    std::cout << "Result: " << (obj.*ptr)(5, 7) << std::endl;
```




```
    return 0; }
```

Program 14:

```
#include <iostream>

int main () {

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

   int* ptr = arr;

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

      std::cout << "Value at index " << i << ": " << *(ptr +
i) << std::endl;

   }

   return 0;

}
```



Program 15:

```
#include <iostream>
int add (int a, int b) {
    return a + b;
}
int main() {
    int (*const ptr)(int, int) = add;
    std::cout << "Result: " << ptr(5, 7) << std::endl;
    return 0;
}
```