

601.220 Intermediate Programming

Initialization and Assignment

Initialization & Assignment

We know there is a difference between `==` and `=`

But there are two kinds of `=`:

- `=` in a declaration, like `int a = 4;` (initialization)
- `=` elsewhere, like `a = 4;` (assignment)

Initialization & Assignment

```
// complex.h
1  #ifndef COMPLEX_H_
2  #define COMPLEX_H_
3  #include <iostream>
4
5  class Complex {
6  public:
7      // Default constructor
8      Complex() : Complex(0.0, 0.0) { std::cout << "Default" << std::endl; }
9      // Non-default constructor
10     Complex(double r, double i) : real(r), imag(i) { std::cout << "Non-default" << std::endl; }
11     // Copy constructor
12     Complex(const Complex& c) : real(c.real), imag(c.imag) { std::cout << "Copy" << std::endl; }
13     // Assignment operator
14     Complex& operator=(const Complex& rhs) {
15         std::cout << "Assign" << std::endl;
16         real = rhs.real;
17         imag = rhs.imag;
18         return *this;
19     }
20
21     double get_real() const { return real; }
22     double get_imag() const { return imag; }
23
24 private:
25     double real, imag;
26 };
27 #endif // COMPLEX_H_
```

Initialization & Assignment

```
// complex_main.cpp
1  #include "complex.h"
2
3  int main() {
4      Complex c;           //default (no-argument) constructor call
5      Complex c2 = {4.9, 0.5}; // = in declaration: *initialization*
6      Complex c3 = c;       // = in declaration; also *initialization*
7      c3 = c2;              // = outside declaration: *assignment*
8      if(c3.get_real() == 4.9) { // == is *equality testing*
9          std::cout << "Real part of c3 is equal to 4.9" << std::endl;
10     }
11     return 0;
12 }

$ g++ -o complex_main complex_main.cpp -std=c++11 -pedantic -Wall -Wextra
$ ./complex_main
Non-default
Default
Non-default
Copy
Assign
Real part of c3 is equal to 4.9
```

Quiz!

What output is printed by the following program?

```
1  #include <iostream>
2
3  int x = 0;
4
5  class Foo {
6  public:
7      Foo() { x++; }
8      Foo(const Foo &obj) { x += 3; }
9      Foo& operator=(const Foo &obj){
10         x += 2; return *this;
11     }
12 };
13
14 int main() {
15     Foo f1;
16     Foo f2(f1);
17     Foo f3 = f1;
18     f3 = f1;
19     std::cout << x << std::endl;
20     return 0;
21 }
```

- A. 0
- B. 4
- C. 8
- D. 9
- E. Some other value is printed

Quiz - answers

What output is printed by the following program?

```
1  #include <iostream>
2
3  int x = 0;
4
5  class Foo {
6  public:
7      Foo() { x++; }
8      Foo(const Foo &obj) { x += 3; }
9      Foo& operator=(const Foo &obj){
10         x += 2; return *this;
11     }
12 };
13
14 int main() {
15     Foo f1;
16     Foo f2(f1);
17     Foo f3 = f1;
18     f3 = f1;
19     std::cout << x << std::endl;
20     return 0;
21 }
```

At line 19:

Symbols (Scope)	Values
x(global)	9