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 and assignment

```
// complex.h:
#include <iostream>

class Complex {
public:
    Complex() : Complex(0.0, 0.0) {
        std::cout << "Default" << std::endl;
    }

    Complex(double r, double i) : real(r), imag(i) {
        std::cout << "Non-default" << std::endl;
    }

    // Copy constructor
    Complex(const Complex& c) :
       real(c.real), imag(c.imag) {
        std::cout << "Copy" << std::endl;
    }

    Complex& operator=(const Complex& rhs) {
        std::cout << "Assign" << std::endl;
        real = rhs.real;
        imag = rhs.imag;
        return *this;
    }

    double get_real() const { return real; }
    double get_imag() const { return imag; }


private:
    double real, imag;
};
```

Initialization and assignment

```cpp
// complex_main.cpp:
#include <iostream>
#include "complex.h"

using std::cout;  using std::endl;

int main() {
    Complex c;   //default (no-argument) constructor call
    Complex c2 = {4.9, 0.5};   // = in declaration: *initialization*
    Complex c3 = c;            // = in declaration; also *initialization*
    c3 = c2;     // = outside declaration: *assignment*
    if(c3.get_real() == 4.9) { // == is *equality testing*
        cout << "Real part of c3 is equal to 4.9" << endl;
    }
    return 0;
}
```

Initialization & assignment

```
Complex c;   //default (no-argument) constructor call
Complex c2 = {4.9, 0.5};   // = in declaration: *initialization*
Complex c3 = c;            // = in declaration; also *initializatio
c3 = c2;     // = outside declaration: *assignment*
if(c3.get_real() == 4.9) { // == is *equality testing*
    cout << "Real part of c3 is equal to 4.9" << endl;
}
```

```
$ g++ -std=c++11 -Wall -Wextra -pedantic -o complex_main complex_main.c
$ ./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?

```
#include <iostream>

int x = 0;

class Foo {
public:
  Foo() { x++; }
  Foo(const Foo &obj) { x += 3; }
  Foo& operator=(const Foo &obj)
    { x += 2; return *this; }
};

int main() {
  Foo f1;
  Foo f2(f1);
  Foo f3 = f1;
  f3 = f1;
  std::cout << x << std::endl;
  return 0;
}
```

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