

02635 Fall 2016 — Module 12 (solutions)

Exercises

1. Write a "Hello World!" program in C++, i.e., a program that writes a string to the standard output. Compile and run your program.

```
#include <iostream>
int main(int argc, const char *argv[]) {
    std::cout << "Hello 02635!" << std::endl;
    return 0;
}
```

2. Write a program that prompts the user to input his or her name and age, and then writes a suitable welcome message to the user. Use a `string` to store the name and use a integer to store the age.

```
#include <iostream>
#include <string>

int main(int argc, const char *argv[]) {
    string name;
    int age;

    std::cout << "Hello, what is your name? ";
    std::cin >> name;
    std::cout << "How old are you? ";
    std::cin >> age;
    if (age < 5 || age > 80)
        std::cout << "Hello " << name << ", impressive "
            << "computer skills for a " << age << "-year-old!\n";
    else if (name.length() > 10)
        std::cout << "Hello " << name << ", that's an unusually long name.\n";
    else
        std::cout << "Hello " << name << "!\n";

    return 0;
}
```

3. Do exercise 11-1 in "Beginning C++".

```
#include "Integer.h"

int main(int argc, const char *argv[]) {

    Integer I1 = Integer();
    I1.set(5);
    I1.print();

    Integer I2 = Integer();
    I2.set(6);
    I2.print();

    Integer I3 = Integer();
    I3.set(-2);
    I3.print();

    // I1.i = 5; // This would result in a compiler error

    return 0;
}
```

See Integer.h and Integer.c at the end of this document.

4. Do exercise 11-2 in "Beginning C++".

```

#include "Integer.h"

int main(int argc, const char *argv[]) {

    Integer I1 = Integer();
    I1.set(5);
    I1.print();

    Integer I2 = Integer(I1);
    I2.print();
    I2.set(6);
    I2.print();

    Integer I3 = Integer();
    I3.print();

    std::cout << "I1.compare_ref(I2):\n";
    int i = I1.compare_ref(I2);
    std::cout << "Return value: " << i << std::endl;

    std::cout << "I1.compare_val(I2):\n";
    i = I1.compare_val(I2);
    std::cout << "Return value: " << i << std::endl;

    return 0;
}

```

See Integer.h and Integer.c at the end of this document.

5. Do exercise 11-3 in "Beginning C++".

```

#include "Integer.h"

int main(int argc, const char *argv[]) {

    Integer I, I4, I5, I6, I7, I8;
    I4.set(4);
    I5.set(5);
    I6.set(6);
    I7.set(7);
    I8.set(8);

    // Compute 4*5^3+6*5^2+7*5+8
    I = I4.multiply(I5.multiply(I5.multiply(I5))).add(
        I6.multiply(I5.multiply(I5))
    ).add(
        I7.multiply(I5)
    ).add(
        I8
    );
    I.print();

    return 0;
}

```

Exercises 3, 4, and 5: Integer.h and Integer.c

```

#ifndef INTEGER_H
#define INTEGER_H

#include <iostream>

class Integer {
private:
    int i;          // value
public:
    Integer();       // "constructors"
    Integer(const Integer& I);
    void set(int newi); // "setter"
    int get();        // "getter"
    void print();     // print value
    int compare_val(const Integer I);
    int compare_ref(const Integer& I);
    Integer add(const Integer& I);
    Integer subtract(const Integer& I);
    Integer multiply(const Integer& I);
};

#endif

```

```

#include "Integer.h"
Integer::Integer() : i(0.0) {
    std::cout << "Creating new Integer object...\n";
}

Integer::Integer(const Integer& I) : i(I.i) {
    std::cout << "Creating new Integer object "
               "using copy constructor...\n";
}

void Integer::set(int newi) {
    std::cout << "Setting value of Integer object to " << newi << std::endl;
    i = newi;
}

int Integer::get() {
    return i;
}

void Integer::print() {

```

```

    std::cout << "Value of integer object: " << i << std::endl;
}

int Integer::compare_val(const Integer I) {
    if (i < I.i)
        return -1;
    else if (i == I.i)
        return 0;
    else
        return 1;
}

int Integer::compare_ref(const Integer& I) {
    if (i < I.i)
        return -1;
    else if (i == I.i)
        return 0;
    else
        return 1;
}

Integer Integer::add(const Integer& I) {
    std::cout << "Adding " << i << " and " << I.i << std::endl;
    Integer sum = Integer();
    sum.set(i + I.i);
    return sum;
}

Integer Integer::subtract(const Integer& I) {
    std::cout << "Subtracting " << I.i << " from " << i << std::endl;
    Integer diff = Integer();
    diff.set(i - I.i);
    return diff;
}

Integer Integer::multiply(const Integer& I) {
    std::cout << "Multiplying " << i << " and " << I.i << std::endl;
    Integer prod = Integer();
    prod.set(i*I.i);
    return prod;
}

```