CISS240: Introduction to Programming
Assignment 2

Name: _____

Objectives

- Declare integer variables
- Input and output of integer values
- Use integer operators

Instructions.

- Your program must be well-written. You must follow the style in your notes as closely as possible. Take note of the spaces and blank lines I used in my examples. Badly written programs will very likely result in a poor grade for this assignment. Points will be taken off for sloppy work.
- It's important to remember this: In your printouts for all assignments, there must be no wraparound.
- All outputs must match exactly the output shown. That includes every single space and every blank line.
- The format of your program must look like this (replacing "smaug" with your name of course):

```
// Name: smaug
// File: a02q01.cpp

#include <iostream>

int main()
{
    *** YOUR WORK HERE ***

    return 0;
}
```

In particular:

- You must have your name and the name of the file at the top of each C++ source file as shown above.
- Your C++ source file must end with a blank line.

- Instructions on uploading your work will be provided in class.

Read the questions carefully before diving in!

Each question contains test cases that your program must pass. For instance for Q1, there are three test cases. Test 1 looks like this:

Test 1

```
Enter w: 1
Enter h: 2
Enter f: 3
IQ: 1
```

This is a complete test run of your program and should be treated as an exact screenshot of your console windows. The underlined text is user input. The above test case means that when you run your program, it should print

```
Enter w:
```

and wait for the user to enter a value. The user enters `1` and press the enter key. Your program then prints

```
Enter h:
```

At this point in Test 1, the user enters `2`. Etc. The program ends after the program's output of `IQ: 1` as shown above. Note that for Q1 there's a Test 2: Test 2

```
Enter w: 6
Enter h: 5
Enter f: 4
IQ: 3
```

This shows you what is expected when you run your program against for a second run. This is not the continuation of the program execution in Test 1 which has already ended.

Note that you should create a new project for each question. For easy maintenance of your assignments, I suggest you have a folder `ciss240` somewhere in your `Documents`, and in that you have a folder `a`, and in folder `a` you have a folder `a02`, and you have solutions folders `a02q01`, `a02q02`, etc. in the folder `a02`:

```
.
.
.
ciss240
|
+--- a
```

```
|
+--- a01
|    |
|    +--- a01q01
|    |
|    +--- a01q02
|
+--- a02
     |
     +--- a02q01
     |
     +--- a02q02
```

Note that the name for the C++ source file for question 1 (i.e. the cpp file in project `a02q01`) must be `a02q01.cpp`, etc.

Q1. According to Elbert Ainstein, your IQ is given by the following formula:

```
IQ = 3 * w / h + (3 + f) / 42
```

where w is your waist (in inches), h is your height (in inches), and f is the number of fingers on your hands. The divisions are all integer divisions (i.e. quotients). Write a program that prompts the user for w, h, and f and print his/her IQ. (No ... the software won't sell.)

Test 1.

```
Enter w: 1
Enter h: 2
Enter f: 3
IQ: 1
```

Test 2.

```
Enter w: 6
Enter h: 5
Enter f: 4
IQ: 3
```

Test 3.

```
Enter w: 10
Enter h: 10
Enter f: 10
IQ: 3
```

Q2. Write a program that prompts the user for his/her age and print his/her age next year. The following are test cases that your program must pass:

Test 1.

```
How old are you? 20
You are 20 years old now. Next year you will be 21.
```

Test 2.

```
How old are you? 155
You are 155 years old now. Next year you will be 156.
```

Test 3.

```
How old are you? -5
You are -5 years old now. Next year you will be -4.
```

Q3. According to Al Chemiz the amount of gold you can produce from x amount of air (cubic feet), y amount of water (gallons), z number of worms, i amount of wood (pounds) boiling at t temperature (celsius) is given by the following formula:

$$x + \frac{y}{z+i}t^3$$

Write a program that prompts the user for x, y, z, i, t and prints the value for the above expression. Assume that the division is an integer division. Your program must pass the following tests:

(Note that when capturing several integer inputs from the keyboard using `std::cin`, you can enter all inputs on one line separated by any whitespace such as a space or a tab.)

Test 1.

```
1 2 3 4 5
1
```

Test 2.

```
5 4 3 2 1
5
```

Test 3.

```
42 41 3 2 1
50
```

Q4. Write a program that waits for a 5-digit number to be entered by the user and then displays the digits of the number entered by the user, with a space separating adjacent digits. The following are test cases that your program must pass.

Test 1.

```
31452
3 1 4 5 2
```

Test 2.

```
97531
9 7 5 3 1
```

Test 3.

```
30000
3 0 0 0 0
```