NAME: _____

Learning Objectives

- Learn about the boolean data type.
- Learn how to use conditional expressions (forms of the "if" statement).
- Learn how to compare integers, floating point numbers and strings.
- Learn how to write loops and use them to solve problems.
- Learn about nested branches and nested loops.
- Learn how to analyze code.

Instructions

Work out the answers to these problems manually without the use of a computer. This will help you develop the ability to read and analyze code.

Another reason to practice solving these problems manually is that problems of this type will appear on Quiz 1 and subsequent quizzes and exams. During the quiz/exam, you will not have access to a computer to solve these problems, so you need to develop the ability to read code, analyze it and think critically about it.

After coming up with answers manually, you can write and run test code to check whether you have solved the problems correctly.

Submit a single document in your repository for this assignment with the answers to these questions. This document should be located in a folder named *a1*.

Problems

```
int i = 0;
while (i < 7)
{
    cout << 2 * i;
    ++i;
}
```

Figure 1

1) When the code in Figure 1 runs, what does it output to the console? (5 points)

```
for (int i = 0; i < 7; ++i)
{
    cout << 2 * i;
}
```

Figure 2

2) Rewrite the code in Figure 2 so that it uses a while loop rather than a for loop to accomplish the same output. (5 points)

```
int k = 100;
for (int i = 0; i < k; ++i)
{
    // This is the body of the for loop.
    cout << i;
}
```

Figure 3

3) How many times does the body of the *for loop* run in the code in Figure 3? (5 points)

```
int i = 11;
while (i <= 99)
{
    i = i + 3;
}
cout << i;
```

Figure 4

4) When the code in Figure 4 runs, what does it output to the console? (5 points)

```
cout << 1 / 2 << ", " << 1 / 2.0;
```

Figure 5

5) When the code in Figure 5 runs, what does it output to the console? (5 points)

```
int i = 2;
int k = i++ * 2;
cout << ++i * 2 + k;
```

Figure 6

6) When the code in Figure 6 runs, what does it output to the console? (5 points)

7) Write a C++ program that computes the sum of integers 0 through n, where n is an integer entered by the user. (5 points)

8) Write a C++ program that prints every number between 330 and 550, inclusive. (5 points)

9) Write a program that prompts the user to enter a number between 3 and 12, inclusive. If the user enters a number inside [3, 12], the problem displays *good number*, otherwise the program displays *bad number*. (5 points)

```
int i;
cin >> i;
if (i % 2 == 1) {
    cout << "odd number"
}
```

10) The code in Figure 10 prints *odd number* when the user enters an odd number and does not print anything when the user enters an even number. Rewrite the code so that it prints *even number* when the user enters an even number and does not print anything when the user enters an odd number. (5 points)

5 of 5