

CSC 2000

Introduction to C++ Programming Language

Assignment 02

50 points

Due 02/05/2020 (11:45 A.M.)

Assignment Objectives:

- . Learn about control structures
- . Examine relational and logical operators
- . Explore how to form and evaluate logical (Boolean) expressions
- . Discover how to use the selection control structures if, if... else, and switch in a program

Answer questions 1 to 5 on a word file; write a program for each of Q.6 - Q.8.

All assignments must be submitted by the Canvas. **No email or hard copy** is accepted. You must follow the following format:

- For non-programming questions, use a word file to type your answers. Don't use the text box on the Canvas to answer the questions or to write comments, we will not read it.
- State your answer clearly.
- For programming questions, include only the source file for each problem.
- Submit your file to the Canvas. You must submit your assignment on time; otherwise, you will receive zero. In addition, you cannot submit your file more than one time.
- There will be several folders on the Canvas. You need to upload your file(s) using the correct folder on the Canvas.
- Name each file: "Assignment Number(Question number(s))".
- To upload your file(s):
 - In Course Navigation, click the Assignments link.
 - Click the title of the assignment.
 - Click the **Submit** Assignment button.
 - Add **File. ...**
 - Add Another **File. ...**
 - Submit** Assignment. ...
 - View **Submission.**

It is your responsibility to make sure that each file is uploaded correctly. If you uploaded a wrong file, you receive zero; files will not be accepted after due date even if you have a prove that the file is created before the due date.

Make sure you review the Cheating & Plagiarism policy on Canvas.

1. (6 points)

Suppose that `x`, `y`, and `z` are `int` variables and `x = 12`, `y = 34`, and `z = 18`. Determine whether the following expressions evaluate to `true` or `false`.

- a. `!(x - y >= 1)`
- b. `z <= 7 || y < 12`
- c. `(x + y != 40) && (x != z)`
- d. `(z - x >= y) || (y - x != z + 4)`
- e. `(5 - x <= 2 * y) && (y - 15 >= z) || (x - 5 != y - 2 * z)`
- f. `(18 / 3 == 2 * 3)`

2. (6 points)

Evaluate the following expressions:

- a. `5 + 6 == 3 + 7`
- b. `2 * 6 - 4 >= 9 - 1`
- c. `'U' >= 't'`
- d. `'A' <= 'a'`
- e. `'#' <= '+'`
- f. `6.28 / 3 < 3 - 1.2`

3. (3 points)

If the number of items bought is less than 5, then the shipping charges are \$7.00 for each item bought; if the number of items bought is at least 5, but less than 10, then the shipping charges are \$3.00 for each item bought; if the number of items bought is at least 10, there are no shipping charges. Correct the following code so that it computes the correct shipping charges.

```
if (numOfItemsBought > 10)
    shippingCharges = 0.0;
else if (5 <= numOfItemsBought || numOfItemsBought <= 10);
    shippingCharges = 3.00 * numOfItemsBought;
else if (0 < numOfItemsBought || numOfItemsBought < 5)
    shippingCharges = 7.00 * numOfItemsBought;
```

4. (4 points)

Suppose that num is an `int` variable. Consider the following C++ code:

```
cin >> num;
if (num >= 0)
    switch (num)
    {
        case 0:
            num = static_cast<int>(pow(num, 3.0));
            break;
        case 2:
            num = ++num;
            break;
        case 4:
            num = num - 4;
            break;
        case 5:
            num = num * 4;
        case 6:
            num = num / 6;
            break;
        case 10:
            num--;
            break;
        default:
            num = -20;
    }
else
    num = num + 10;

cout << num << endl;
```

- a. What is the output if the input is 5?
- b. What is the output if the input is 26?
- c. What is the output if the input is 2?
- d. What is the output if the input is -5?

5. (4 points)

Suppose that beta is an `int` variable. Consider the following C++ code:

```
cin >> beta;
switch (beta % 7)
{
    case 0:
    case 1:
        beta = beta * beta;
        break;
    case 2:
        beta++;
        break;
    case 3:
        beta = static_cast<int>(sqrt(beta * 1.0));
        break;
```

```

case 4:
    beta = beta + 4;
case 6:
    beta = beta--;
    break;
default:
    beta = -10;
}
cout<< beta<< endl;

```

- a. What is the output if the input is 11?
- b. What is the output if the input is 12?
- c. What is the output if the input is 0?
- d. What is the output if the input is 16?

6. (10 points)

The roots of the quadratic equation $ax^2 + bx + c = 0$, $a \neq 0$ are given by the following formula:

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

In this formula, the term $b^2 - 4ac$ is called the **discriminant**. If $b^2 - 4ac = 0$, then the equation has a single (repeated) root. If $b^2 - 4ac > 0$, the equation has two real roots. If $b^2 - 4ac < 0$, the equation has two complex roots. Write a program that prompts the user to input the value of a (the coefficient of x^2), b (the coefficient of x), and c (the constant term) and outputs the type of roots of the equation. Furthermore, if $b^2 - 4ac \geq 0$, the program should output the roots of the quadratic equation. (*Hint*: Use the function `pow` from the header file `cmath` to calculate the square root. Chapter 3 explains how the function `pow` is used.)

7. (8 points)

Write a program that mimics a calculator. The program should take as input two integers and the operation to be performed. It should then output the numbers, the operator, and the result. (For division, if the denominator is zero, output an appropriate message.) Some sample outputs follow:

```

3 + 4 = 7
13 * 5 = 65

```

8. (9 points)

Let l be a line in the x - y plane. If l is a vertical line, its equation is $x = a$ for some real number a . Suppose l is not a vertical line and its slope is m . Then the equation of l is $y = mx + b$, where b is the y -intercept. If l passes through the point (x_0, y_0) , the equation of l can be written as $y - y_0 = m(x - x_0)$. If (x_1, y_1) and (x_2, y_2) are two points in the x - y plane and $x_1 \neq x_2$, the slope of the line passing through these points is $m = (y_2 - y_1) / (x_2 - x_1)$. Write a program that prompts the user to enter two points in the x - y plane. The program outputs the equation of the line and uses `if` statements to determine and output whether the line is vertical, horizontal, increasing, or decreasing. If l is a nonvertical line, output its equation in the form $y = mx + b$.