***CSC 3020***

***Java Programming***

**Lab 02**

**25 points**

**Due 10:10 A.M.**

Assignment Objectives:

■■ To combine conditions using logical operators (**!**, **&&**, **||**, and **^**).

■■ To implement selection statements.

■■ To use static methods in the **Character** class.

■■ To represent strings using the **String** class.

■■ To write programs for executing statements repeatedly using loop.

■■ To write nested loops.

■■ To process characters in a string using a loop.

■■ To define methods with formal parameters.

■■ To declare, create, and initialize an array.

■■ To develop and invoke methods with array arguments and return values.

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

1. Submit your file to the Canvas. You must submit your file on time; otherwise, you will receive zero.
2. You can upload your file as many times as you like. Only the last attempt counts because the last file you uploaded is the only file your instructor will see.
3. There will be several modules on the Canvas. You need to upload your file using the correct module on the Canvas.
4. Name the lab file: *Lab (labt number)*
5. To upload your file(s):

* In Course Navigation, click the ASSIGNMENTS module.
* Click the title of the assignment.
* Click the **Submit** Assignment button.
* Add **File**. ...
* **Submit** Assignment. ...
* View **Submission**.

*It is your responsibility to make sure that the 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.***

**Solution to this assignment will not be posted on Canvas; however, any question can be discussed in the class upon request of a student.**

Write a program that will read a line of text. Display all the letters that occure in the text, one per line and in alphabetical order, along with the number of times each letter occurs in the text. Use an **array** of base type int of length 26, so that the element at index 0 contains the number of a’s, the element at index 1 contains the number of b’s, and so forth, Allow both upperCase and lower Case. Define a **method** that takes a character as an argument and returns an int value that is the correct index for that character. For example, the argument ‘a’ results in 0 as the return value, the argument ‘b’ gives 1 as the return value, and so on. Allow the user to **repaet** this task until the user says she/he is through.

Sample output:

