Lab 6: Text Processing
UNK CSIT 150 Object Oriented Programming

Objectives
• Practice parsing a string
• Practice using the Character wrapper class
• Practice using StringBuffers

General Requirements

In this lab, you must write your code following the proper programming style. The bottom line is:

• Use *indentation* to show the logical structure of your code
• Use blank lines to separate code blocks and give each code block a comment
• Give documentation comment to the class and each method

Programming Practice

Imagine that you are developing a software package that requires users to enter their own passwords. Your software requires that users' passwords meet the following criteria:

- The password should be at least six characters long.
- The password should contain at least one uppercase and at least one lowercase letter.
- The password should have at least one digit and one special character. (A special character is not alphabetic and is not a digit).
- The password should not be in the array of not allowed passwords. In general, this array may include several words, or be null. For testing purposes, do not allow passwords to be one of the following strings: `"P@ssword1","Myp@ssw0rd".`

Ask for the user name and password. Keep asking for a password until the user enters a password that meets the criteria above.

Output the username and the password. Then, output the user and the hidden password "******". Add JavaDoc comments to this program.

Encrypt the password as follows:

- Replace each character with the ASCII character that follows it. (If the character is 'a', replace it with 'b'.)
- Reverse the entire password.

Output the username and original password and output the username and its encrypted version.

More Bonus 5 points: ask the user to enter an encrypted password, and using a decrypt method, verify if it is the original password or not.

Report to the Instructor

Before you leave, show the instructor

• The source code
• How the code runs
• The generated technical documents

Lab 6: Password

Name(s): _____

Evaluation

Requirement	**Possible Points**	**Points Received**	**Comments**
Check for invalid passwords in array. The sample invalid passwords provided are just a sample file In general, this array may include several words, or be null.	2		
Check for digits	1		
Check for upper case	1		
Check for lowercase	1		
Check for non-alphabetic	1		
Check for length	1		
Loop until valid	1		
Username/asterisks	2		

Programming style	**Possible Points**	**Points deducted**	
Inconsistent indentation	**-1**		Indent at least 3 spaces inside each brace
Poor use of white space	**-1**		Too many or too few lines between statements
Heading documentation, includes programmer names, date, algorithm, basic purpose	**-2**		
All sources not cited (Remember to cite all code used, even class demo code.)	**-1**		
JavaDocs not used on each method	**-1**		
Poor variable names	**-1**		Should not start with uppercase. No single letter variables.
Poor structure/logic issues	**-2**		
Program specifications			
Bonus:			