CSI 1390  Assignment 4  Fall 2012

Instructions:

1. The assignment is due before 11:59 PM on Sunday, November 25th, 2012.
2. This assignment is to be solved INDIVIDUALLY.
3. For each question you have to write a Java program, name each question as A1QY, where Y is the number of the question, paste your java code to .doc file and zip all your files in A1_xxxxxx.zip, where xxxxx is your student number, and submit it through the Virtual Campus.
4. Assignments submitted after November 25th will be marked with 15% penalty. Submissions after November 26th will receive a mark of 0.

Question 1 (50 marks):

Write a program that will read a line of text that ends with a period, which serves as a sentinel value. Display all the letters that occur 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 lowercase letters as input, but treat uppercase and lowercase versions of the same letter as being equal.

Hints: use one of the methods toUpperCase or toLowerCase in a wrapper class Character. You will find it helpful to 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. Note that you can use a type cast, such as (int)letter, to change a char to an int. Of course, this will not get the number you want, but if you subtract (int)'a', you will then get the right index. Allow the user to repeat this task until the user says she or he is through.

Question 2 (50 marks):

Traditional password entry schemes are susceptible to "shoulder surfing" in which an attacker watches an unsuspecting user enter their password or PIN number and uses it later to gain access to the account. One way to combat this problem is with a randomized challenge-response system. In these systems the user enters different information every time, based on a secret in response to a randomly generated challenge. Consider the following scheme, in which the password consists of a five-digit PIN number (00000 to 99999). Each digit is assigned a random number that is 1, 2, or 3. The user enters the random numbers that correspond to their PIN instead of their actual PIN numbers. For example, consider an actual PIN number of 12345. To authenticate the user would be presented with a screen such as:

PIN: 0 1 2 3 4 5 6 7 8 9
NUM: 3 2 3 1 1 3 2 2 1 3

The user would enter 23113 instead of 12345. This doesn't divulge the password even if an
attacker intercepts the entry because 23113 could correspond to other PIN numbers, such as 69440 or 70439. The next time the user logs in, a different sequence of random numbers would be generated, such as:

PIN: 0 1 2 3 4 5 6 7 8 9
NUM: 1 1 2 3 1 2 2 3 3 3

Write a program to simulate the authentication process using this scheme. Store an actual PIN number in your program. The program should use an array to assign random numbers to the digits from 0 to 9. Output the random digits to the screen, input the response from the user, and output whether or not the user's response correctly matches the PIN number.