EECS 1510: Object Oriented Programming

Project 5 – Methods, Arrays, and Strings

150 Points Due in class Thursday March 27, 2014

Problem 2 correction - The corrected return type of the function is highlighted in yellow.

Submission Procedure. For the submission of this project, be sure to read the posted file "Printed Submission of Projects.doc".

For the printout of each program, give ***two*** sample runs, one using the values from a sample dialog below, the other using values from some "special" case. Your program must also follow the same style guidelines used for Project 4.

The material for this assignment is Chapters 5 and 6, as well as sections 9.1 - 9.3.

Program 1. ScanningText.java (50 points) Write a program to read in a line of text and give as output the number of letters, the number of string tokens, the length of the longest token, and a table giving the frequency of each letter. For example,

```
Enter a single line of text:
Now is the time for all good men to come to the aid of their country.

The line contains 53 letters.
The line contains 16 string tokens.
The longest token has 7 characters.
The frequency of letters is
A -- 2
C -- 2
D -- 2
E -- 6
F -- 2
G -- 1
H -- 3
I -- 4
L -- 2
M -- 3
N -- 3
O -- 9
R -- 3
S -- 1
T -- 7
U -- 1
W -- 1
Y -- 1
```

The table of letter frequencies must use capital letters (as above), but can be in more than one column. Use an array to store the letter frequencies. Name the program **ScanningText.java**.

Hint: Read the line of text into a string variable using the nextLine method,

```
       String line;
       line = input.nextLine();
```

and later use

```
       int ch, count;
```

```
      for (ch='a'; ch<='z'; ch++){
         -- Count the number of occurrences of ch in the line
         -- Print the count if > 0
      }
```

Program 2 (40 points) BinaryConversion.java

Consider the following dialog:

```
Enter a binary number: 1110
Conversion to decimal:  14
Enter a binary number: 10010000
Conversion to decimal:  144
Enter a binary number: -1
All set !
```

Write the application implied above. In particular, the program will read a sequence of binary strings and convert each one to a decimal integer. The program will terminate when the string -1 is given. You must do the conversion by hand, and NOT use the predefined functions in Java for the wrapper class Integer.

You must use a function for the conversion to decimal, where the parameter to the function is the binary string, and the return value is the equivalent decimal integer.

```
      public static int binaryToDecimal (String binaryString)
```

Program 3 (30 points) Reverse.java

Write a program that reads in a series of positive integers termінted by a -1, e.g.

```
      73  95  61  21  90  85  14  78 -1
```

The values should be stored in an array. The program then prints the values in reverse order as well as the average (to one decimal place). For example,

```
      Please enter the integers: 73 95 61 21 90 85 14 78 -1
      The values in reverse order are
            78   14   85   90   21   61   95   73
      The average is 517/8 = 64.6
```

You must use a function a function for computing the reverse, passing the array as an argument. Thus the function header will be similar to

```
      void reverse(int[] A)
```

Program 4 (30 points) Valid Phone Numbers

Phone number can have one of several valid formats: In particular, strings like

```
      419-460-1212     (419)460-1212     460-1212
```

are valid but strings like

```
      419-460     (419)460-a321     46012-12
```

are not. Write a program to read in a string and check whether it has the format

```
      ddd-ddd-dddd
```

where each d is a digit.