CS101- Algorithms and Programming I
Lab 04

Lab Objectives: `while` loops, nested `while` loops.

For all labs in CS 101, your solutions must conform to the CS101 style guidelines (rules!)

For the following questions, you should only use `while` loops in your solutions.

1. Write a program, `Lab04_Q1.java` that inputs words from the user, until the user enters 'exit'. When the user enters 'exit' the program should display the word that comes *second* alphabetically. Your program should not be case sensitive.

   Sample Runs:

```
Enter strings (exit to stop):
zebra
cat
dog
ant
bunny
exit
String that comes second alphabetically: bunny

Enter strings (exit to stop):
house
exit
Not enough input data...

Enter strings (exit to stop):
house
exit
Not enough input data...

Enter strings (exit to stop):
house
mouse
house
exit
Not enough input data...

Enter strings (exit to stop):
house
car
house
exit
String that comes second alphabetically: house
```

2. Write a program, `Lab04_Q2.java` that inputs a string from the user and displays the length of the longest block of *letters* with the same case. I.e., the longest block of uppercase *letters* or lowercase *letters*.

   Input string: HELLO therehow AREYOU TODAY -> longest block is 8
   Input string: hi how is your day? -> longest block is 4
   Input string: 52343 3424 -> longest block is zero

3. Write a program, `Lab04_Q3.java` that inputs a `long` binary number and displays its decimal and hexadecimal values.
   Notes:
   - You should not use `String` methods in your solution.
   - 4 binary digits is called a *nibble*, each nibble (or partial nibble) corresponds to a hex digit. To convert to hex you should process the binary number in nibbles.
   - Binary-Decimal-Hex Conversion Information.
   - **Hint:** the characters A – F are represented in ascii by the decimal values 65 – 70. You can cast an `int` value as a `char`: `char ch = (char) 65 //‘A’`

   Sample Run:

```
Enter binary number (exit to quit): 1011010000110000
              Binary             Decimal         Hexadecimal
    1011010000110000               46128               #B430

Enter binary number (exit to quit): 0110100
              Binary             Decimal         Hexadecimal
              110100                  52                 #34

Enter binary number (exit to quit): 0000
              Binary             Decimal         Hexadecimal
                   0                   0                  #0

Enter binary number (exit to quit): 1111
              Binary             Decimal         Hexadecimal
                1111                  15                  #F

Enter binary number (exit to quit): exit
```