CIST 2371 Introduction to Java

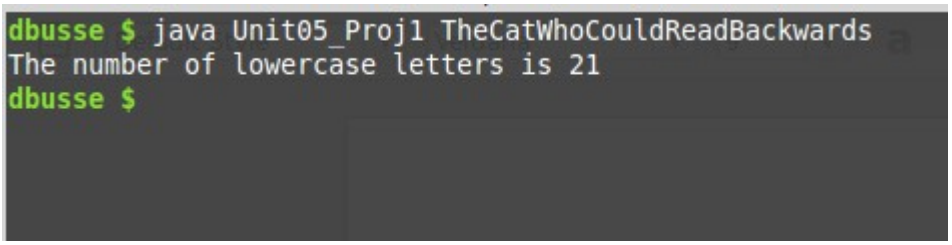
Unit 05 Lab

Due Date: _____

Create a folder named Unit05 and put all the files in this folder.

Part 1

For this part create a program named **Unit05_Proj1.java** that will accept one argument from the command line, which will be a string (you can't get anything else, right?). The program will then check each character in that string and determine if it is an lowercase character or not and keep a count. When it hits the end of the String it will print out how many lowercase characters there are. Here is a screen shot of a test run.



White space between words on the command line separate arguments. So if you type in:

```
>java Unit05_Proj1 The Fox Jumped
```

There are three arguments and the first argument is "The".

Part 2

Design, by creating a UML diagram, a class named MyInteger. The class contains:

- An int data field named value that stores the int value represented by this object.
- A constructor that creates a MyInteger object with the passed int value.
- A get method that returns the int value.
- The instance methods isEven(), isOdd(), and isPrime() that return true if the value in this object is even, odd, or prime, respectively.
- The static methods isEven(int), isOdd(int), and isPrime(int) that return true if the specified value is even, odd, or prime, respectively.
- The static methods isEven(MyInteger), isOdd(MyInteger), and isPrime(MyInteger) that return true if the specified value is even, odd, or prime, respectively.
- The instance methods equals(int) and equals(MyInteger) that return true if the value in this object is equal to the specified value.
- A static method parseInt(char[]) that converts an array of numeric characters to an int value.
- A static method parseInt(String) that converts a string into an int value.

Create a UML diagram that shows the MyInteger class, its attributes and methods. The UML diagram should be drawn using the UMLet drawing tool. The diagram should be exported as an image and then imported into a word processing document as demonstrated in class. The document should have your name, course assignment and date at the top like this:

Sally Doe
CIS 2371 Summer 2014
Unit 04 Lab
June 12, 2014

The file should be an PDF formatted file and be named Unit05_<first initial><lastname>.pdf, for example Unit05_sdoe.pdf. Any word processor can save or export to PDF.

Once your UML diagram is complete implement the class naming the Java file **MyInteger.java**. Write a program named **MyIntegerTester.java** that tests all methods in the class. One technique is to write both classes at the same time. As much as possible you implement and test one method at a time. This minimized the amount of new code you have to look at when debugging a method. Only the "tester" class will have main() method in it.

What To Turn In
You will zip up the Unit05 folder containing your source code files and your UML diagram document into a zip file named Unit05_<your first initial><your lastname>.zip. For example Tom Swift's zip file would look like this: Unit05_tswift.zip

Be sure to test this file before turning it in. Copy it into a temp folder, unzip it and try to compile and run the files in that temp folder.

Once you are satisfied that your zip file is OK, turn it in via the ANGEL drop box for this Unit.

The Comment Block
EVERY source code file you turn in for this course must have the comment block handed out the first day of class (customized for the situation) or you will get no credit (kind of like forgetting to put your name on your term paper – bad move).

Rubric (50 points total)
P1 Unit05_Prog1 has comment block 2pts
P1 Unit05_Prog1 gets input from the command line 8pts
P1 Unit05_Prog1 produces correct output by using String and other methods 10pts
P2 UML Diagram – WP document with correctly drawn diagram 10pts
P2 MyInteger class has comment block 2pts
P2 MyInteger class has all methods 6pts
P2 MyIntegerTester has comment block 2pts
P2 MyIntegerTester tests all methods in MyInteger class and they work 10pts