COMP 1230
LAB-4 (Files and Exception Handling)
Due: June 13, 2017

This week you will have an opportunity to work with both file streams and exceptions.

a) **[10 Points]** In this program, you have to implement a generic Stack with the following public interface:

`+ push(E e): void`	Adds an element at the topmost position of the stack.
`+ pop(): E`	Removes and returns the topmost element from the stack
`+ peek(): E`	Returns the topmost element from the stack without removing it.
`+ isFull(): boolean`	Returns true if the stack is full, false otherwise.
`+ isEmpty() Boolean`	Returns true if the stack is empty, false otherwise
`+ Stack(int c)`	Constructor to initialize the stack with initial capacity
`+ Stack()`	Default constrictor initializes the stack with a default capacity of 10.

You must use java generic and collection framework in your implementation, otherwise you will not get any points.

Write a client code that will test different features of your stack class and submit that client code as well.

Save the source code in “Stack.java” and the client code in “StackClient.java”.

b) **[20 Points]** Your task is to write a program that uses a textfile containing any number of lines of integer values as input. Each line in the file is supposed to have no more than 15 values in it. If the file exists and each line in the file does in fact contain 15 or fewer integers, then the program processes and reports on each line in turn by displaying its line number, the number of values on that line, and the range of values on the line.

But, as you know, things are not always as they seem, or as they are supposed to be. For example, the file your program is looking for may not be there, there may be no integers at all on some lines, some lines may contain more values than the allowed maximum of 15, and some lines may contain

"values" that aren't even integers (i.e., badly formatted "invalid" integer values). The program must recognize and respond appropriately to each of these problems.

Note: You must implement exception handling. Not only am I requiring it, but Java itself will also require it since you will have to use Java resources that force you to deal with the `IOException`. In addition to this exception, you are also required to handle the `ArrayIndexOutOfBounds` and the `NumberFormatException` at appropriate points in your program.

Your program should read in, from the command line, the name of a text file containing any number of lines of integers and then process each line in turn. Processing a line means computing and then reporting, for that line, its line number, along with the range of values on the line.

The program should recognize a number of problems that may occur when trying to process such a file. First, the file itself may not be present. If so, the following message is displayed:

```
filename.txt (the system cannot find the file specified).
Program will now terminate.
```

If the file exists, and contains lines of integers, possibly with blank lines, possibly lines with too many values, and possibly lines with invalid integers on them, then the following messages are output, as appropriate:

```
Line #: Number of values = # and value range = [#, #].
Line #: This is a blank line.
Line #: This line contains more than the maximum of 15 allowed data
values.
Line #: This line contains an invalid integer value.
```

For example, if a file named “text.txt” contains the following text:

```
1 2 30

3 4 5
2 3 4 5 6.5 7
abs sej jdh& 4
1 2 3 4 5 6 6 6  8 99    0 0 -1
    34 34       23
```

```
24
1 2 3 4 5 6 7 8   9 10 -2 3 4 5 6 7
1
```

Then the output should be like this:
Output:

```
The file test.txt was successfully opened.
The data in it will now be processed.
Press Enter to continue ...
Line 1:Number of values = 3 and value range is [1, 30].
Line 2:This is a blank line.
Line 3:Number of values = 3 and value range is [3, 5].
Line 4:This line contains an invalid integer value.
Line 5:This line contains an invalid integer value.
Line 6:Number of values = 13 and value range is [-1, 99].
Line 7:Number of values = 3 and value range is [23, 34].
Line 8:Number of values = 1 and value range value range consists of
this single value.
Line 9:This line contains more than the maximum of 15 allowed data
values.
Line 10:Number of values = 1 and value range value range consists of
this single value.
```

Submission:

- Zip all source code files in a folder (lab4_yourname) and submit the zip file through blearn.