

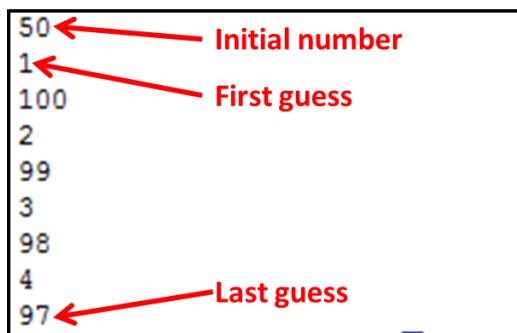
CPSC 233: Assignment 1 (*Due February 1 at 4 PM*)

New learning concepts: Input and output, variables and constants, documentation, branching, and loops in Java. Note: you can write this as a non-Object Oriented solution with only a single class and the main method. Aside from ‘main’ you cannot implement other static methods. (Don’t worry if you don’t know the significance of the word ‘static’ yet, an explanation will come later in the term).

Description

Write a program that implements the logic of a simple guessing game. To speed up marking the values of the initial number and guesses **MUST** come from an input file (and not as prompts to the user). Figure 1 shows the contents and format of an example input file.

Contents of file “lose.txt”



```
50
1
100
2
99
3
98
4
97
```

Figure 1: Sample input file (game is ‘lost’)

The user will enter the name of the input file in the form of a command line argument (e.g., “java Guessing lose.txt”). All the remaining information needed by the program will come from this file. The format of the file is plain text and each number will be an integer that resides on its own line. The first line contains the initial number (the number ‘picked’ by the computer). All the lines that follow will contain the ‘guesses’. The numbers can range from between 1 –100 (inclusive of both end points). If the guess matches original number then the game has been won. A suitable congratulatory message should be displayed and the program will end. Otherwise the program should give ‘hints’ if the guess is higher or lower than the initial number. The game is lost if eight guesses have been made without any of them matching the initial number. When the game is lost a suitable consolation message should be displayed and the program will end (Figure 2 illustrates a sample run of the program using “lose.txt” as the input file).

Sample execution using “lose.txt”

```
[csb assignment1 23 ]> java Guessing lose.txt
Higher!
Lower!
Higher!
Lower!
Higher!
Lower!
Higher!
You ran out of tries. Sorry you lose the game!
```

First command line
argument (name of the
input file)

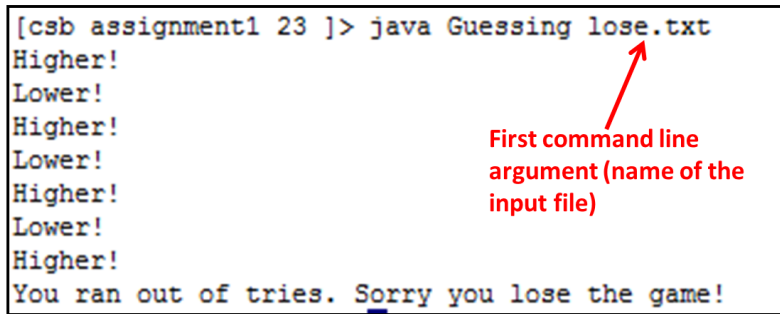


Figure 2: Sample program execution using 'lose.txt' as the input file.

Since file input and command line arguments haven't been covered extensively yet you will be provided a minimal starting program that handles those parts of the assignment. (Your course instructor will let you know how you can access the file). Note: the starting program handles file input only; it does not play the game (because you are responsible for writing that part of the program yourself).

You can assume that the input file will contain exactly eight guesses. Unless the input file contains a scenario where the game is lost the program should terminate after reaching the correct guess (the other guesses will be ignored by the program if the game is won).

External libraries/methods that can be used (you don't write yourself)

You can use pre-created code for console input and output (e.g., 'print', class 'Scanner'), classes for file input and to methods convert from String to numeric types.

Important reminders

- **Handing in your assignment:** Use the UNIX 'submit' program: "submit -c 233 -a 1 Guessing.java"
- **Collaboration:** Assignments must reflect individual work, group work is not allowed in this class nor can you copy the work of others.