University of Windsor

School of Computer Science - Winter 2018
60-214: Languages, Grammars, and Translators
Assignment-1
Due: February 2, 2017 @11:59pm

Objective:

The objective of this assignment is to get familiar with regular expression and practice Java programming. A useful Java Regular Expressions link is given here and also posted in Blackboard.

Assignment Specification:

The task is to count the number of user defined identifiers (variables, user defined functions, labels etc) in programs written in *C* programming language. A useful *C* Programming link is given here and also posted in Blackboard. Here you can assume that there are no comments in the programs. The input will have multiple lines. Find out the identifiers from a text file (e.g., `A11-input.txt`) and print the output into a text file (e.g., `A11-output.txt`). The output file should contain a line like `"distinct/unique identifiers: 7"`. **Do not write anything else into the output file!**

Some sample test cases/input files (posted in Blackboard) and output are given below:

`A11-input.txt, A12-input.txt, A13-input.txt, A14-input.txt, A15-input.txt`

`distinct/unique identifiers: 7, 8, 3, 1, 3`

Write a program in one java file only. Your program should be able to run by typing:

```
javac A1_yourSID.java
java A1_yourSID A11-input.txt
```

In this assignment, the input file name will be provided as the argument of the program. The output should be written in a file called `"A11-output.txt"` for `"A11-input.txt"` as an input. **Do not use keyboard input!**

Identifiers:

An identifier is a series of characters consisting of letters, digits and underscores (_) that does not begin with a digit. *C* is case sensitive—uppercase and lowercase letters are different in *C*, so $a1$ and $A1$ are different identifiers.

Submission Instructions:

Submit your source file `A1_yourSID.java` in BlackBoard. For example, if your SID is `"123456789"`, then the submission file is `"A1_123456789.java"`. You can submit multiple times but the mark will be posted based on your latest attempt. Comment your program carefully so that it can be read and understood. If your program is not properly commented you may lose marks. See **marking scheme** for details.

Marking Scheme:

Program name:	5
Comments/Code documentation:	10
Complete program, no compile/logical error, and correct output:	35