Lab 11

Objective

The objective of this lab is to practice the following concepts:

1- POSIX Basic Regular Expressions

For reference, see the slides and this website (make sure to use the POSIX Basic Regular Expression (BRE) syntax and not the Extended Regular Expression (ERE)).

Overview

Use grep to search for specific lines in a c program utilizing Basic Regular Expressions (BRE). The grep command should be executed as grep <BRE> <filename> where <BRE> is the required regular expression and <filename> is a file name. Two files are given for testing: log3.c and log3.py. Download the two files and apply the grep commands you will write to them. Some tasks will involve the two files (search for patterns in both files simultaneously), in this case, the grep command must be run as **grep <BRE> <file1> <file2>**.

Tasks

1. Write a grep command that will look for single-line comments in the both files (log3.py and log3.c).
   Note:
   a- Python single-line comments can start with '#' while C single-line comments start with "//".
   b- Lines that start with "#include" or "#define" should not be matched.
   c- The matched single-comments should start with line; lines has code and end with a single-single line comment should not be matched.
   d- To simplify the BRE, you can use Linux pipelining to run two grep commands back-to-back where the matched lines by the first grep command are passed as input to the second grep command lines.
   e- You can use the -v option in grep to exclude lines that match the BRE instead of including them. For example, you can have the output from the first grep command match all lines that start with '#' or "//" and then exclude those lines that start with "#include" or "#define" in the second grep command.

2. Write a grep command that will look for lines that import/include libraries in both files (log3.py and log3.c). These are lines that start with “import” in Python or “#include” in C. For Python, any import statement should be matched. For C, however, only lines that include a standard library header file should be matched. Therefore, the line “#include “utils.h”” should not be matched.
3. Write a grep command that will match lines that contain an if statement in both files (log3.py and log3.c).
4. Write a grep command that will match lines that contain any assignment operator (e.g. =, /=, +=, etc) but not relational operators (e.g. ==, <=, etc) in both files (log3.py and log3.c).
5. Write a grep command that matches a variable declaration of the types (int, double, or long long) in the C file. Arrays declaration should also be matched, so the line: long long powers[MAX+1]; should be matched.

Submission

Submit one pdf file named lab11.pdf which contains the 5 grep commands in text and a screenshot for each command demonstrating the output. The output of the grep command are lines that the corresponding question asks to match. Only the required lines are matched. For example, if a regular expression is looking for a numerical variable declaration, lines that don’t contain numerical variable declarations should not be matched.