Assignment 1

CS3310 Compiler Design Lab
due August 20, 2015 at 23:55 on moodle

Implement a language for network visualization. The input is a specification of representing connections, and manipulating them. The output would be a graphviz-readable file that can be manually viewed.

Instructions

1. Download CS13B000.tar.gz to a directory, go to the directory in a terminal
2. tar xvzf CS13B000.tar.gz
3. mv CS13B000 CSxxB0xx   # use your roll number
4. cd CSxxB0xx
5. make   # compile given demo program
6. make test   # run it on testcases
7. make tar   # generate .tar.gz for submission

Details

1. Use C language only, and lex-yacc tools.
2. There are only three relations: friendof, classmateof and roommateof.
3. You have to consider transitivity for roommateof and classmateof (i.e. if a roommateof b and b roommateof c then a roommateof c). However, friendof relation is not transitive.
4. None of the relations are reflexive. Thus, there should be no relation anil R anil.
5. All the relations are symmetric (a friendof b means b friendof a, and others). Therefore, your final graph must be undirected.
6. For each edge, a label mentioning the relationship must be given. No colors should be used.
7. If required, you may assume that there are maximum 100 people in the whole program.
8. If required, you may assume that there are maximum 10 nestings (if and foreach).
9. There are ten public testcases along with their expected answers. Check your program on the testcases using: *make test*. The answers are obtained manually, so in case you find any issue, please post on moodle. Ten more private testcases would be used for evaluation. Each testcase carries 0.5 marks.
10. The scope of the variables used in the *foreach* construct is restricted to the loop alone; that is, the variable's value is not available outside the loop body. But each *foreach* will use a separate variable. Thus, you will not have a nested statement as below:

```
foreach $x ...
   foreach $x ....
      ...
   endforeach
endforeach
```

The following is also not expected, and will not be present in testcases.

```
foreach $x ...
         ...
endforeach
foreach $x ...
         ...
endforeach
```

11. For erroneous testcase, your program should print ERROR\n as the only output. Empty lines and arbitrary whitespace (space and tabs) are allowed anywhere.
12. Note that *foreach* syntax contains $x rather than x for the variable name.
13. Print the output (graphviz graph) on stdout. If you are writing to a file, simply add system(“cat filename”); to your program. Do not worry about the order of the edges in the output, the evaluation script should take care.
14. We will have penalty for wrong submission. All your efforts can go waste if you don't follow the instructions.

Submission

Submit a single tar file as ROLLNO.tar.gz which extracts to directory ROLLNO which contains all the relevant lex-yacc files, and Makefile . You can use the following command to create it: *make tar*

After the submission, it is your responsibility to download your own .tar.gz from moodle and see if it extracts properly. If there are issues, resubmit on moodle. If for some reason, there are issues with moodle, you have to make sure your send your assignment to either the instructor or any of the TAs before 23:55 on the submission date. We will not entertain requests such as “Sir, I uploaded wrong file” or “I forgot to include one file in the archive”.