Indian Institute of Technology Tirupati

CS3191 Compiler Design Laboratory

Lab #1 Due Date: Aug 13, 2019, 5pm

Lab Objectives

- To understand the developmental aspects of compiler.
- To learn the tools, namely Lex and Yacc used in the development of compilers.

Lab Outcomes

- Ability to write regular expressions for different language constructs.
- Ability to develop simple tools for scanning and parsing.
- Ability to develop a miniature compiler.

Exercises

Objective: Learning Lex through simple tasks.

1. Write a lex program to read a string of digits of arbitrary length from **a file** and verify whether the given string of digits is a binary, octal, decimal, hexa decimal number or not a valid number.
   Test cases:

input	**output**
01000001	binary
010901	decimal
90878	hexa
010007	octal
010909	decimal
acbd	hexa
0109	decimal
klfw	invalid
abcd	hexa

2. Write a lex program to read a string of digits of arbitrary length from **a file** and verify whether given string is Positive/Negative, Integer, Integer with Exponent form, Real, Real with Exponent form.
   Example: 23, 23, +23, 12e3, 12e3, 12e+3, 12e23, 2.3, 2.3, 3.14E2, 3.12e+2, 3.12e14, 3.13E23.
   Test cases:

input	output
12	positive integer
0.10	positive real
10.01	positive real
stp	neither integer nor real
0.2.3	neither integer nor real
2..3	neither integer nor real
.23	real
12e3	positive integer in exponential form
12.2e-3	positive real in negative exponential form

3. Write a lex program to read a string of digits of arbitrary length from **a file** and verify whether given string is a valid IPv4 address or not

Test cases:

input	output
123.234.43.2	valid
123.234.43.2.4.3	invalid
257.890.123.978	invalid
0.0.0.0	valid

4. Write a lex program to check whether given String is valid Email id or not. Specifications: Email id has **username** and **domainname**. Domainname may contain subdomains and can have a depth of any number of levels. Username and domainname may contain digits, alphabets and special characters, underscore (_) and/or dot (.). However username or domainname should not start with underscore and the special characters should not appear consecutively either in domainname or in username. Leading dot and trailing dot in username/domainname is not allowed.

Sample Structure: username@domainname

Test cases:

input	output
kishan@gmail.com	valid email
kanduru@iittp.ac.in	valid email
kishan123_23@yahoo.co.in	valid email
kishan.gmail.com	invalid email
kishan.@gmail.com	invalid email
.kishan@yahoo.com	invalid email
_____@gmail.com	invalid email
@%@&*(@gmail.com	invalid email
kishan..kanaut@yahoo.com	invalid email
kishan.kandukuru@yahoo..co..in	invalid email
kishan@abc.gov	valid email
kishan@xyz.abc.gov	valid email
kishan@pqr.xyz.abc.gov	valid email
kishan@x.pqr.xyz.abc.com	valid email
@gmail.com	invalid email
kishan@	invalid email
123890@gmail.com	valid email

Submission guidelines

All Exercises should be submitted in the following format.

- All files and folders should be lowercase letters
- Create a folder with name yourrollnumber_week_1 (say cs17b001_week_1) and create subfolders, namely 1, 2,.., n for each problem given for the first lab, in this single folder
- Prepare a separate lex and make files for each of the exercise problems
- Makefile should generate final executable file named **scanner**
- The input must be given through a file and the file name should be taken through command line arguments
- Copy the lex and make files into their respective sub folders
- Create a Readme file in the main folder
- Dont keep any unrealed or executable files
- Finally tar and compress the yourrollnumber_week_1 directory as yourrollnumber_week_1.tar.gz and upload the same to the course page at Moodle before the due date.

Any kind of copying, sharing code with others, and Malpractices attract high penalties to the extent of referring to the **Institute Level Disciplinary committee**.

Evaluation is done based on

- Output
- Logic (wherever applicable)
- Naming convention, code readability, comments etc.,
- Adherence to the instructions