

### Compiler Lab. 2013 (List 1: Pre-mid Sem.)

1. Generate a hand held lexical analyzer by writing a program in C/C++/Java to tokenize the simple “Hello World” program of C language. First of all, define the tokens for this program. The original program should necessarily include few comments. Display the tokens by removing all the comments, i.e., the comments should be ignored for tokenization.
2. Write a simple lexical analyzer using C/C++/Java to detect the lexical errors in a trivial C program containing declaration of int, float, char variables and simple if-else statement.
3. Using C/C++/Java, read command line input and identify the digits and alphabets from it. Read the same from a file and process. Repeat both the parts using lex/flex.
4. Write a Program in Flex which identifies verbs, articles, adverbs from a sentence. Define the set of at least 10 verbs, articles and adverbs each. Write the code in C language to identify verbs only.
5. (a) Write a Program in Flex which identifies C integers and float numbers. Your program should respond to various inputs as follows:

Input	Output
234	Integer
-765	Integer
76549	Out of Integer Range
-98.213	Float
8234.0	Float

(b) Write a program in Flex which can accept:

(i) Even number of zeroes and ones. The order of occurrence of 0's and 1's is not important. For Example: Your program should accept 00, 11, 1100, 00001111, 1010, 11000101.

(ii) Four digit Numbers divisible by 5.

6. Consider following grammar

```
stmt → if expr then stmt | if expr then stmt else stmt | ε
expr → term relop term | term
term → id | num
relop → < | <= | > | >= | <>
id → letter(letter | digit)*
num → digit+(.digit+)
letter → a | b | c | ... | z | A | B | ... | Z
digit → 0 | 1 | 2 | ... | 9
```

Based on the grammar recognize various if then else statements including nested statements. Generate and display the tokens and check for the errors in given input, if any. Use lex/flex or C/C++/Java for implementation of lexical analyzer and parser.

7. (i) Write a Program in Flex which can identifies keywords, operators (Arithmetic, Relational, Logical & Assignment), identifiers, parenthesis, comments, braces etc. The Flex Program should take input from a C file. The output should be generated in following order, based on the order of occurrence of tokens.

int	Keyword
(	Left parenthesis
=	Assignment Operator
{	Brace open
int	Keyword
a09	Identifier
	Logical Operator
10	Integer
12.451	Float
for	Keyword

(ii) Implement the above problem in either C/C++/Java.

8. Identify the comments in a given C program and the remove all the comments from it. Show the program without comments as output. Use lex/flex or C/C++/Java for the programming.