Department of Computer Science and Engineering

Course Code: CSE 430 Course Title: Compiler Design Lab

Lab Exercise 1

Read the instructions carefully and submit the exercise within time

Write a program in C/C++ for lexical analysis which takes input from file or keyboard and specify each word or character into the tokens given below. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments and identify duplicate identifiers. Your Program should include a scanner (optional: Input Preprocessor) and a tokenizer.

- Any word either combination of characters and digits or combination of characters: **Identifier**
- Any number : **Constant** ● Single character tokens:
    - **Parenthesis** : ( ), { }, [ ]
    - **Punctuation signs**: ;(semicolon) , : (colon) , , (comma) ○ **Arithmetic operators** : + , -, *, /
- **Logical Operator** : >, >=, <, <=, ==, !=
- **Keyword**: There are total 32 keywords in C. They are:

auto	break	case	char	const	continue	default	do
double	else	enum	extern	float	for	goto	if
int	long	register	return	short	signed	sizeof	static
struct	switch	typedef	union	unsigned	void	volatile	while

Sample Input (Console Input/ File Input):

```
void main() { int
a, b, c; //comment
int a = b*c +   10;
}
```

Sample output:

```
Keyword (2): void, int
Identifier (4): main, a, b, c
Arithmetic Operator (3): =, *, +
Constant (1): 10
Punctuation (2): , , ; Parenthesis
(2): {,}
```