COMPILER DESIGN LAB

WEEK 5 (7.1.18) - EXERCISE

SET - A

1. Write the Lex Program to find the token and its count from the given

```
       c = ++a   +   ++b
```

2. Identify the tokens in the given input statement

```
       print  ( 3 + x *2)
       def f(x): return 3
```

3. Append the next matched string to the current value of the yytext rather than replacing the contents of the yytext. Handle it for Uppercase and Lowercase

4. Match any string of one or more characters that do not include lower case a-z.

5. Comments consisting of string surrounded by /* and */ without intervening */, unless it is inside double-quotes(")
   Input:

```
      {
         while ( (c = input()) == '*' )
                 ;
            if ( c == '/' )
            break;    /* found the end */
            }
                        while (c ! = eof)
                {
                if (c = =1 )
                        return "YES";   /* Returns YES  */
                else
                        return "NO";    /* Returns NO  */
            }
```

SET - B

1. Write the Lex Program to find the token and its count from the given

```
            c = a++    +    ++b
```

2. Identify the tokens in the given input statement

```
          def f(x):
          if x >= 1:
            return x * x
          else:
            return x
          print 3
```

3. Identify the keywords and convert it into Uppercase.

4. Retain three initial characters in the yytext and returns the remaining characters to the input stream.

5. All strings of lowercase letters in which the letters in are in ascending lexicographic order.