Aneesh Nangal
2019CSB1071

CS - 202
Written Assignment

(1) The lexical syntax can be given as :-

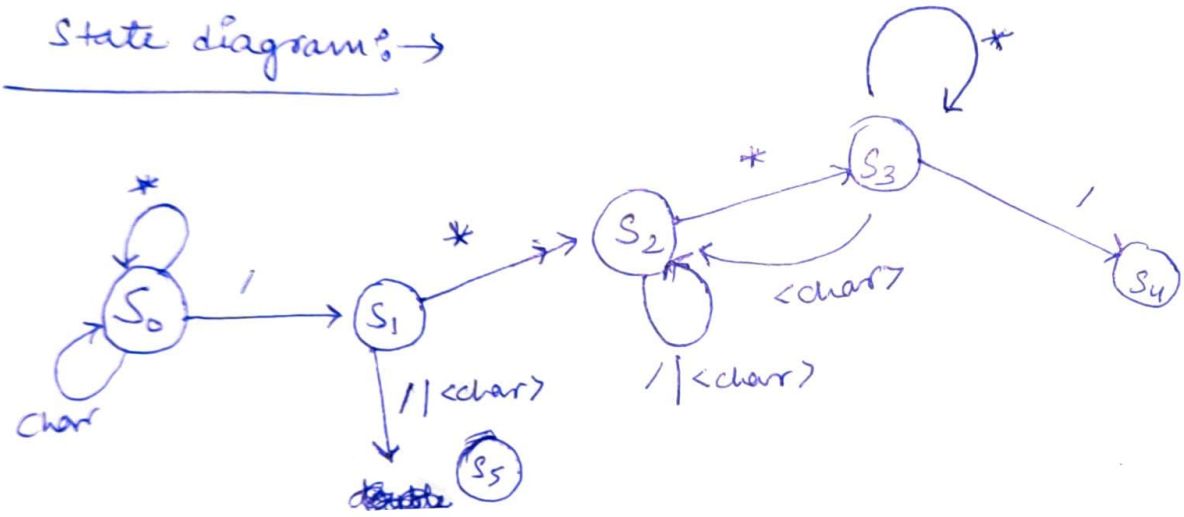
<comment> → /*{(<char>) | *<star> | <slash>/)} */

<char> → all characters except * & /

<star> → *|<char>

<slash> → /K char>

State diagram :→



S_0 :- Comment has not begun

S_1 :- / detected, it might be a comment

S_2 :- Comment initialisation.

S_3 :- * detected after initialisation of comment

S_4 :- Comment complete

S_5 :- Not a comment

2)

```
void if_stmt () {
  if (<next_token> != <if_code>) { error (); }
  else {
     lex();

     if ( next_token != left_parent ) {error (); }
     else {
     lex ();
     boolexpr();
     if (next_token != right_parent) {error (); }
     else {
     lex();

     statement();
     if (next_token == else_code ) {
        lex();
        statement ();

     }
     }
     }
  }
}
```

(3) There are 35 tokens in the given C code

1) identifier 'main'
2) l-paren '('
3) r-paren ')'
4) l-brace '{'
5) int 'int'
6) star '*'
7) identifier 'u'
8) comma ','
9) star '*'
10) identifier 'v'
11) comma ','
12) identifier 's'
13) semi ';'
14) identifier 'u'
15) equal '='
16) amp '&'
17) identifier 's'
18) semi ';'
19) identifier 'v'
20) equal '='
21) amp '&'
22) identifier 's'
23) semi ';'
24) identifier 'printf'
25) l-paren '('
26) string-literal '"%d %d"'
27) comma ','
28) identifier 's'
29) comma ','
30) star '*'
31) identifier 'u'
32) r-paren ')'
33) semi ';'
34) r-brace '}'
35) eof ''