14/09/2021 CD LAB - 02 SACHIN RAGHULT

OBSERVATION 2019103573

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

```
%option noyywrap
%{
  #include<stdio.h>
%}
%%
[A-Z]+[\t\n\.\,] { printf("matched string:%s\n",yytext);}
%%
int main()
{
   printf ("Enter the string : ");
   yylex();
   return 0;
}
```

Output:

```
Enter the string : Write a PROGRAM printing HELLOWORLD
matched string : PROGRAM
matched string : HELLOWORLD
```

2) Match one or more occurences ab concatenated

```
%option noyywrap
%{
 #include<stdio.h>
%}
%%
(ab)+ [\t\n\.\,] {printf("matched string : %s\n", yytext); }
.    ;
%%
int main()
{
  printf("Enter the string : ");
  yylex();
  return 0;
}
```

Output:

```
Enter the string : apple ab ant abab aeroplane
matched string : ab
matched string : abab
```

3) Write the lex program to find the token and its count from input the file for following.

```
%option noyywrap
%{
  #include<stdio.h>
  int identifier = 0, number = 0, key = 0, white = 0;
   assign = 0, operatorsym = 0, splchar = 0;
%}
%%
if | then | int | float | real | else | while | for
  { key++. printf("keywords : %s\n", yytext); }
```

```
[a-z][a-z 0-9]* { identifier++, printf("Identifier: %s\n", yytext); }
[0-9]* { number++, printf("number : %s\n", yytext); }
[+,*,/,-] { operatorsym++, printf("operator symbol: %s\n", yytext); }
:={assign++, printf("Assignment operator: %s\n", yytext); }
[!@#$%^&*,()] { splchar++, printf("splcharacter: %s\n", yytext); }
[\t\n" "] { white++, printf("White spaces: %s\n", yytext); }
.
;
%%
int main()
{
    yylex;
    printf("keyword : %d\n", key);
    printf("Identifier : %d\n", identifier);
    printf("Number : %d\n", number);
    printf("white spaces : %d\n", white);
    printf("Assignment operator : %d\n", assign);
    printf("operator symbol : %d\n", operatorsym);
    printf("special character : %d\n", splchar);
    return 0;
}
```

Output:

① float x = 32.68

Keywords : float
White spaces :
Identifier : x
number : 32
number : 68
white spaces :

② int num1 = 10000

Keywords : int
white spaces :
Identifier : num1
number : 10000
white spaces :