Name:- Sumit Kumar Yadav
Roll No. :- 18CS30042

Assignment - 2

① (a) L = [b-d f-h j-n p-t v-z]

L: only containing consonants (no vowels)

String → $(L|a)^* (L|e)^* (L|i)^* (L|o)^* (L|u)^*$

(b)

String → $[(aa+bb)^*b] + [(aa+bb)^*(ab+ba)][a(bb)^*a + b(aa)^*b + (ab+ba)(aa+bb)^*(ab+ba)][a(bb)^* + ab(aa+bb)^*b + ba(aa+bb)^*b]$.

(c) $S \rightarrow b^* (a(\epsilon|b))^*$

(d) string → $b^* a^* (\epsilon|b) a^*$

② /* Regular Expression ~~Def~~ Declaration */

```
INT        "int"
FLOAT      "float"
DOUBLE     "double"
RETURN     "return"
ID         [A-Z a-z][A-Z a-z 0-9]*
WS         [ \t \n]
```

Scanned with CamScanner

```
/* Translation Rules */
%%
{INT}        {Printf("<KEYWORD, int >\n");}
{FLOAT}      {Printf("<KEYWORD, float >\n");}
{DOUBLE}     {Printf ("<KEYWORD, double>\n"); }
{RETURN}     {Printf ("<KEYWORD,  return>\n"); }
{ID}         {Printf("<IDENTIFIER, %s >\n", yytext); }
"="          {Printf("<OPERATOR , =>\n"); }
"+="         {Printf (" <OPERATOR, +=>\n");}
" /="        {Printf ("<OPERATOR, /=>\n");}
" ^"         {printf ("<OPERATOR, ^> \n"); }
" , "        {Printf (" <SPECIAL-~~OPERATO~~CHARACTER, ,>\n");}
" ; "        { Printf ("<SPECIAL_CHARACTER, ;>\n");}
" ( "        {Printf("< SPECIAL_CHARACTER, (>\n");}
" )"         {Printf("< SPECIAL- CHARACTER,) >\n");}
" { "        {Printf (" <SPECIAL - CHARACTER, {} >\n"); }
" }"         {Printf(" <SPECIAL_ CHARACTER, } >\n"); }
  {WS}           /* white - Space rule */
  {PUNC}     {Printf ("<PUNCTUATION, ; > \n"); }
%%
```

TOKEN Generated :-

<KEYWORD, float>

<IDENTIFIER, Function2Calculate >

<SPECIAL_CHARACTER , (>

<KEYWORD, int >

<IDENTIFIER .a>

<SPECIAL_CHARACTER, ,>

< KEYWORD . double>

<IDENTIFIER , b>

<SPECIAL_ CHA~~EF~~RACTER, ,>

<KEYWORD, float>

<IDENTIFIER , c>

<SPECIAL-CHARACTER,) >

<SPECIAL_CHARACTER, { >

< IDENTIFIER ,b>

<OPERATOR , = >

<IDENTIFIER, b >

<OPERATOR, ^>

<IDENTIFIER, c >

<SPECIAL- CHARACTER, ;>

<IDENTIFIER ,a>

<OPERATOR, +=>

<IDENTIFIER, b>

<SPECIAL_CHARACTER , ;>

<IDENTIFIER, c>

<OPERATOR, /=>

<IDENTIFIER, a>

<SPECIAL_CHARACTER, ;>

<KEYWORD, return>

<IDENTIFIER, c>

<SPECIAL_CHARACTER, ;>

<SPECIAL_CHARACTER, }>

(3) Features :-

(i) Indentation : for python, 4 space indentation is a very important part and it should be tokenized separately.

(ii) In python, lines do not end with a semicolon.

(iii) Apart from these, other parts are tokenized similar to the C language token rules and definition.

"def" : Keyword

func : identifier

a, b, c : identifier

"(", "(", ":", "," : special character

"=", "+" : operators

"return" : Keyword.