20181CSE0621
Sai Ram
7-CSE-10

Part-B

Q.4] • A lexical analyzer by name yylex() must be provided.
• Using lex to produce yylex() is a common choice.
• It produces tokens consisting of a token name and its associated attribute value.

Q.5]

```
%{
#include <ctype.h>
#include <stdio.h>
#define YYSTYPE double
%}
%token NUMBER
%left '+' '-'
%left '*' '/'
%right UMINUS
%%
lines : lines expr '\n' {printf("%g\n", $2);}
      | lines '\n'
      | /* empty */
      ;
expr : expr '+' expr   { $$ = $1 + $3; }
     | expr '-' expr   { $$ = $1 - $3; }
     | expr '*' expr   { $$ = $1 * $3; }
     | expr '/' expr   { $$ = $1 / $3; }
     | '(' expr ')'    { $$ = $2; }
     | '-' expr %prec UMINUS { $$ = -$2; }
     | NUMBER
     ;
%%
```

2018ICSE0621
Sai Ram

classmate
Date
Page

```
yylex() {
    int c;
a] while ((c=getchar()) == ' ');
  if ((c == '.' || isdigit(c)))
  {
    ungetc(c, stdin);
    scanf("%lf", &yylval);
    return NUMBER;
  }
  return c;
}
```

b]
~~for ((c=0; c!=' ';)~~
```
for (c=0; c.getchar() != ' '; c++)
{
    ungetc(c, stdin);
    scanf("%lf", &yylval);
    return NUMBER;
}
    return c;
}
```

c]
```
if (c.getchar() != ' ')
{
    ungetc(c, stdin);
    scanf("%lf", &yylval);
    return NUMBER;
}
    return c;
}
```