SPCC LAB 08

icg.l

```
%{
#include"icg.tab.h"
extern char yyval;
%}

%%

[0-9]+ {yylval.symbol=(char)(yytext[0]);return NUMBER;}
[a-z] {yylval.symbol= (char)(yytext[0]);return LETTER;}
. {return yytext[0];}
\n {return 0;}

%%
```

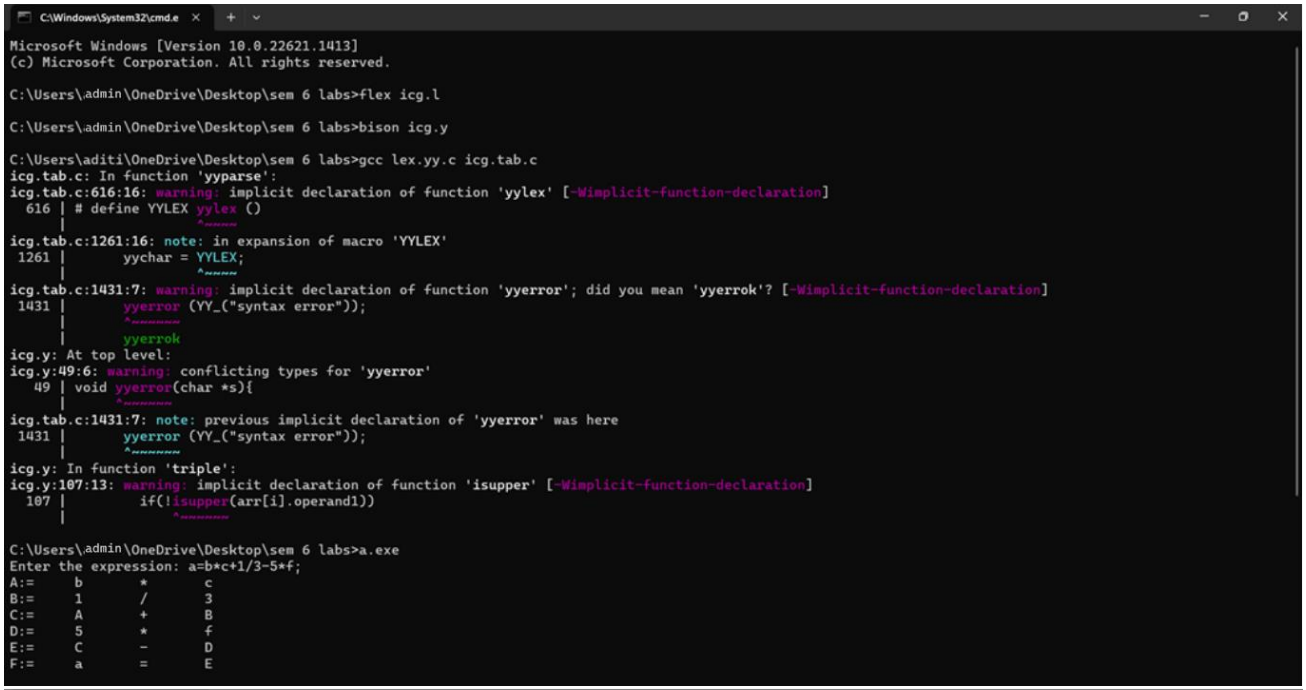
icg.y

```
%{
#include<stdio.h>
char addtotable(char,char,char);

int index1=0;
char temp = 'A'-1;

struct expr{

char operand1;
char operand2;
char operator;
char result;
};

%}

%union{
char symbol;
}


%left '+' '-'
%left '/' '*'

%token <symbol> LETTER NUMBER
%type <symbol> exp
%%

statement: LETTER '=' exp ';' {addtotable((char)$1,(char)$3,'=');};
exp: exp '+' exp {$$ = addtotable((char)$1,(char)$3,'+');}
   |exp '-' exp {$$ = addtotable((char)$1,(char)$3,'-');}
```

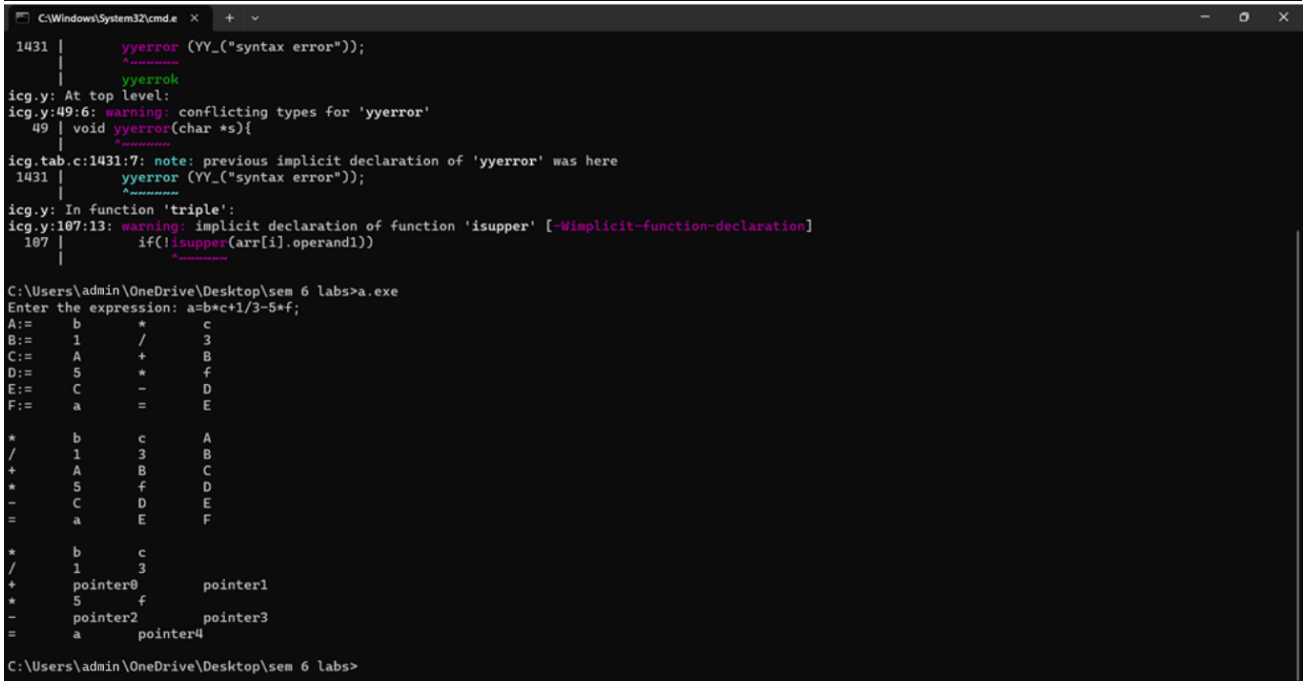
```
    |exp '/' exp {$$ = addtotable((char)$1,(char)$3,'/');}
    |exp '*' exp {$$ = addtotable((char)$1,(char)$3,'*');}
    |'(' exp ')' {$$= (char)$2;}
    |NUMBER {$$ = (char)$1;}
    |LETTER {(char)$1;};

%%

struct expr arr[20];

void yyerror(char *s){
   printf("Errror %s",s);
}

char addtotable(char a, char b, char o){
   temp++;
   arr[index1].operand1 =a;
   arr[index1].operand2 = b;
   arr[index1].operator = o;
   arr[index1].result=temp;
   index1++;
   return temp;
}

void threeAdd(){

   int i=0;
   char temp='A';
   while(i<index1){
      printf("%c:=\t",arr[i].result);
      printf("%c\t",arr[i].operand1);
      printf("%c\t",arr[i].operator);
      printf("%c\t",arr[i].operand2);
      i++;
      temp++;
      printf("\n");
   }
}

void fouradd(){
   int i=0;
   char temp='A';
   while(i<index1){
      printf("%c\t",arr[i].operator);
      printf("%c\t",arr[i].operand1);
      printf("%c\t",arr[i].operand2);
      printf("%c",arr[i].result);
      i++;
      temp++;
      printf("\n");
   }
```

```
}

int find(char l){
   int i;
   for(i=0;i<index1;i++)
      if(arr[i].result==l) break;
   return i;
}

void triple(){
   int i=0;
   char temp='A';
   while(i<index1){
      printf("%c\t",arr[i].operator);
      if(!isupper(arr[i].operand1))
      printf("%c\t",arr[i].operand1);
      else{
         printf("pointer");
         printf("%d\t",find(arr[i].operand1));
      }
      if(!isupper(arr[i].operand2))
      printf("%c\t",arr[i].operand2);
      else{
         printf("pointer");
         printf("%d\t",find(arr[i].operand2));
      }
      i++;
      temp++;
      printf("\n");
   }

}

int yywrap(){
   return 1;
}
int main(){
   printf("Enter the expression: ");
   yyparse();
   threeAdd();
   printf("\n");
   fouradd();
   printf("\n");
   triple();
   return 0;
}
```

Output



```
C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Users\admin\OneDrive\Desktop\sem 6 labs>flex icg.l

C:\Users\admin\OneDrive\Desktop\sem 6 labs>bison icg.y

C:\Users\aditi\OneDrive\Desktop\sem 6 labs>gcc lex.yy.c icg.tab.c
icg.tab.c: In function 'yyparse':
icg.tab.c:616:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declaration]
  616 | # define YYLEX yylex ()
      |                ^~~~~
icg.tab.c:1261:16: note: in expansion of macro 'YYLEX'
 1261 |       yychar = YYLEX;
      |                ^~~~~
icg.tab.c:1431:7: warning: implicit declaration of function 'yyerror'; did you mean 'yyerrok'? [-Wimplicit-function-declaration]
 1431 |       yyerror (YY_("syntax error"));
      |       ^~~~~~~
      |       yyerrok
icg.y: At top level:
icg.y:49:6: warning: conflicting types for 'yyerror'
   49 | void yyerror(char *s){
      |      ^~~~~~~
icg.tab.c:1431:7: note: previous implicit declaration of 'yyerror' was here
 1431 |       yyerror (YY_("syntax error"));
      |       ^~~~~~~
icg.y: In function 'triple':
icg.y:107:13: warning: implicit declaration of function 'isupper' [-Wimplicit-function-declaration]
  107 |         if(!isupper(arr[i].operand1))
      |             ^~~~~~~

C:\Users\admin\OneDrive\Desktop\sem 6 labs>a.exe
Enter the expression: a=b*c+1/3-5*f;
A:=     b       *       c
B:=     1       /       3
C:=     A       +       B
D:=     5       *       f
E:=     C       -       D
F:=     a       =       E
```



```
C:\Windows\System32\cmd.e
 1431 |       yyerror (YY_("syntax error"));
      |       ^~~~~~~
      |       yyerrok
icg.y: At top level:
icg.y:49:6: warning: conflicting types for 'yyerror'
   49 | void yyerror(char *s){
      |      ^~~~~~~
icg.tab.c:1431:7: note: previous implicit declaration of 'yyerror' was here
 1431 |       yyerror (YY_("syntax error"));
      |       ^~~~~~~
icg.y: In function 'triple':
icg.y:107:13: warning: implicit declaration of function 'isupper' [-Wimplicit-function-declaration]
  107 |         if(!isupper(arr[i].operand1))
      |             ^~~~~~~

C:\Users\admin\OneDrive\Desktop\sem 6 labs>a.exe
Enter the expression: a=b*c+1/3-5*f;
A:=     b       *       c
B:=     1       /       3
C:=     A       +       B
D:=     5       *       f
E:=     C       -       D
F:=     a       =       E

*       b       c       A
/       1       3       B
+       A       B       C
*       5       f       D
-       C       D       E
=       a       E       F

*       b       c
/       1       3
+       pointer0        pointer1
*       5       f
-       pointer2        pointer3
=       a       pointer4

C:\Users\admin\OneDrive\Desktop\sem 6 labs>
```