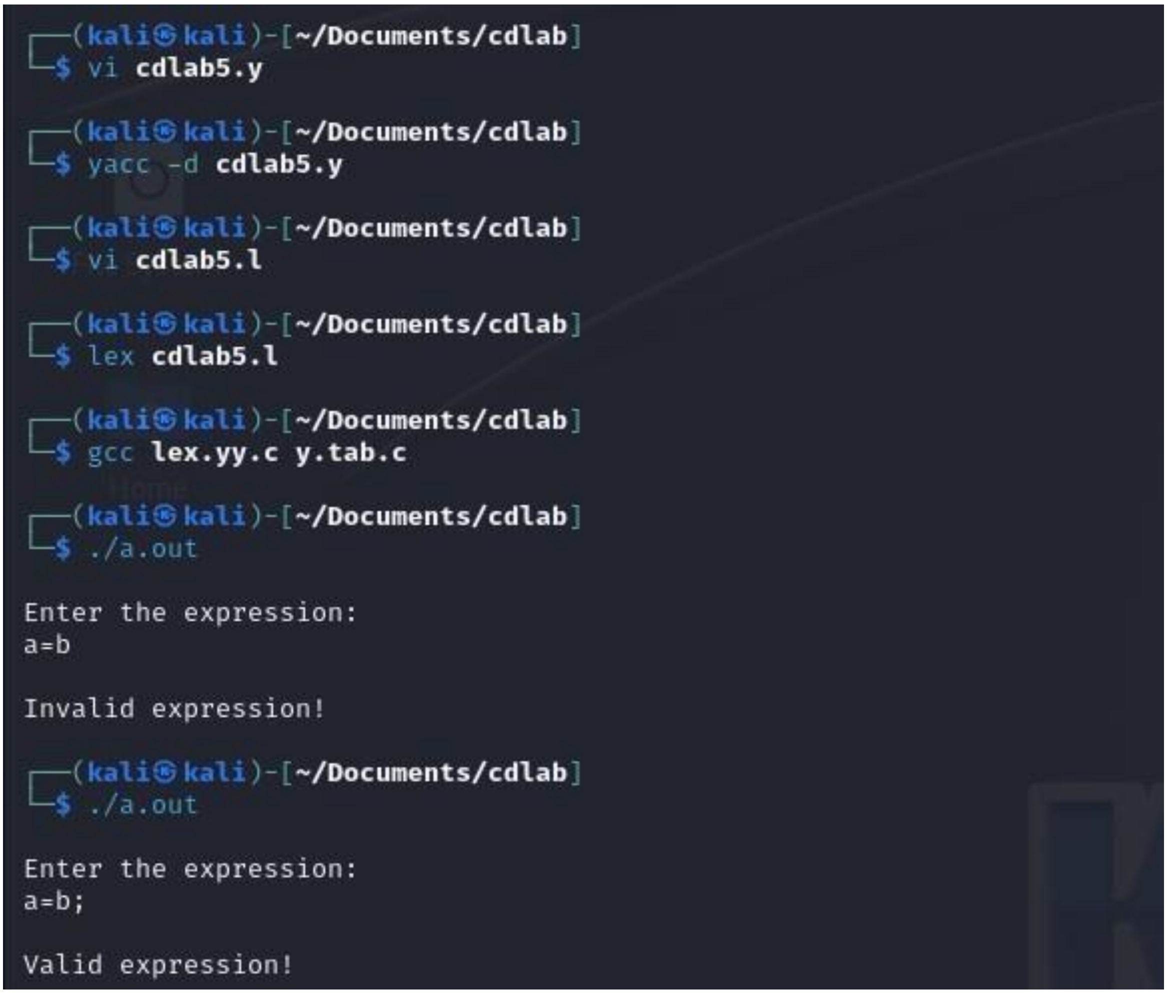
```
int yyerror(){     valid=0;
printf("\nInvalid expression!\n");     return
0;
} int main(){      printf("\nEnter the
expression:\n");     yyparse();
if(valid){         printf("\nValid
expression!\n");
    }}
```

OUTPUT:



```
┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ vi cdlab5.y

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ yacc -d cdlab5.y

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ vi cdlab5.l

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ lex cdlab5.l

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ gcc lex.yy.c y.tab.c

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ ./a.out

Enter the expression:
a=b

Invalid expression!

┌──(kali㉿kali)-[~/Documents/cdlab]
└─$ ./a.out

Enter the expression:
a=b;

Valid expression!
```

RESULT:

Thus, a program to check whether the arithmetic expression using lex and yacc tool is implemented.

Ex No: 6
Date:

RECOGNIZE A VALID VARIABLE WITH LETTERS AND DIGITS USING LEX AND YACC

AIM:

To recognize a valid variable which starts with a letter followed by any number of letters or digits.

ALGORITHM:

Lex (exp6.l):

1. Recognizes letters, digits, any single character, and newline.
2. Returns tokens for letters, digits, and single characters.
3. Indicates the end of input with yywrap().

Yacc (exp6.y): 1. Includes headers and defines global variables.

2. Declares tokens digit and letter.
3. Defines grammar rules for identifiers.
4. Handles syntax errors with yyerror().
5. The main function, obtain the input, parses it, and prints if it's recognized as an identifier.

PROGRAM: exp6.l: %{

```
  #include "y.tab.h"
%}
%%
[a-zA-Z_][a-zA-Z_0-9]* return letter;
[0-9]                  return digit;
.                   return yytext[0];
\n                   return 0;
%%
int yywrap(){ return
1;
```

```
}
```

exp6.y:

```
%{
   #include<stdio.h>
int yylex();     int
yyerror();     int valid=1;
%}

%token digit letter


%%


start : letter s
s :     letter s
      | digit s


      |


      ;
%%
int yyerror(){     printf("\nIts not a identifier!\n");


   valid=0;     return 0; }  int main() {
printf("\nEnter a name to test for an identifier: ");
yyparse();     if(valid) {         printf("\nIt is a
identifier!\n");
   } }
```