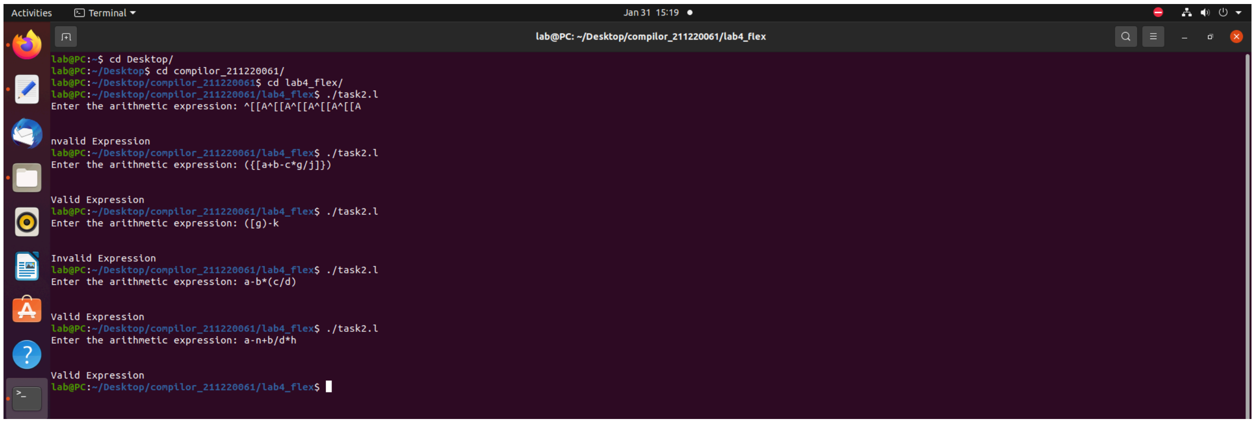
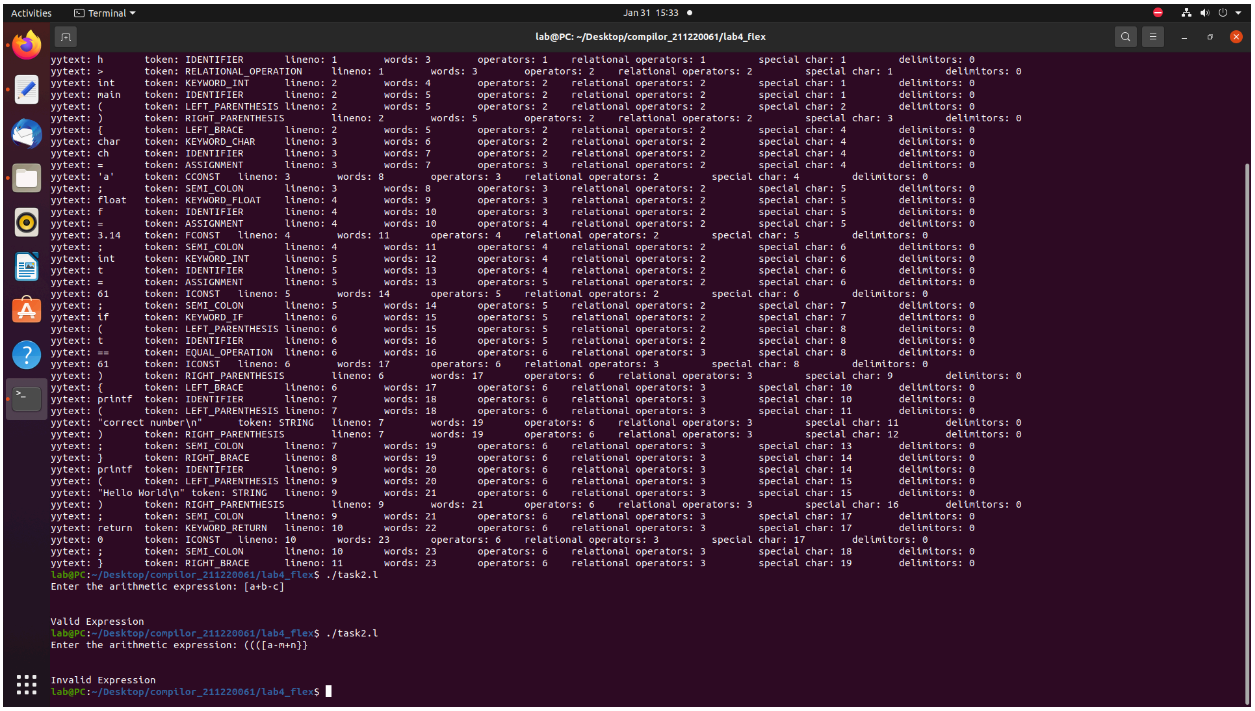
```
// Name: AHAN BANDYOPADHYAY
// Roll No.: 211210008
// CSB353 Compiler Design Lab4

#include<stdio.h>
int main(){
	char ch = 'a';
	float f = 3.14;
	int t = 61;
	if(t==61){
		printf("correct number\n");
	}
	printf("Hello World\n");
	return 0;
}
```

```
%{
#include <stdio.h>
%}
%%
"int"|"float"|"char"|"if"|"else"|"while"|"for"|"return" {
printf("KEYWORD: %s\n", yytext);
}
[{};,()] {
printf("SEPARATOR: %s\n", yytext);
}
[+-/*%=] {
printf("OPERATOR: %s\n", yytext);
}
[a-zA-Z][a-zA-Z0-9]* {
printf("IDENTIFIER: %s\n", yytext);
}
[0-9]+ {
printf("NUMBER: %s\n", yytext);
}
.|\n { /* Ignore other characters */ }
%%
int main() {
yylex();
return 0;
}
```



```
lab@PC:~$ cd Desktop/
lab@PC:~/Desktop$ cd compilor_211220061/
lab@PC:~/Desktop/compilor_211220061$ cd lab4_flex/
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: ^[[A^[[A^[[A^[[A^[[A


nvalid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: ({[a+b-c*g/j]})


Valid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: ([g)-k


Invalid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: a-b*(c/d)


Valid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: a-n+b/d*h


Valid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ 
```



```
yytext: h       token: IDENTIFIER       lineno: 1       words: 3        operators: 1    relational operators: 1         special char: 1         delimitors: 0
yytext: >       token: RELATIONAL_OPERATION     lineno: 1       words: 3        operators: 2    relational operators: 2         special char: 1         delimitors: 0
yytext: int     token: KEYWORD_INT      lineno: 2       words: 4        operators: 2    relational operators: 2         special char: 1         delimitors: 0
yytext: main    token: IDENTIFIER       lineno: 2       words: 5        operators: 2    relational operators: 2         special char: 1         delimitors: 0
yytext: (       token: LEFT_PARENTHESIS lineno: 2       words: 5        operators: 2    relational operators: 2         special char: 2         delimitors: 0
yytext: )       token: RIGHT_PARENTHESIS        lineno: 2       words: 5        operators: 2    relational operators: 2         special char: 3         delimitors: 0
yytext: {       token: LEFT_BRACE       lineno: 2       words: 5        operators: 2    relational operators: 2         special char: 4         delimitors: 0
yytext: char    token: KEYWORD_CHAR     lineno: 3       words: 6        operators: 2    relational operators: 2         special char: 4         delimitors: 0
yytext: ch      token: IDENTIFIER       lineno: 3       words: 7        operators: 2    relational operators: 2         special char: 4         delimitors: 0
yytext: =       token: ASSIGNMENT       lineno: 3       words: 7        operators: 3    relational operators: 2         special char: 4         delimitors: 0
yytext: 'a'     token: CCONST   lineno: 3       words: 8        operators: 3    relational operators: 2         special char: 4         delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 3       words: 8        operators: 3    relational operators: 2         special char: 5         delimitors: 0
yytext: float   token: KEYWORD_FLOAT    lineno: 4       words: 9        operators: 3    relational operators: 2         special char: 5         delimitors: 0
yytext: f       token: IDENTIFIER       lineno: 4       words: 10       operators: 3    relational operators: 2         special char: 5         delimitors: 0
yytext: =       token: ASSIGNMENT       lineno: 4       words: 10       operators: 4    relational operators: 2         special char: 5         delimitors: 0
yytext: 3.14    token: FCONST   lineno: 4       words: 11       operators: 4    relational operators: 2         special char: 5         delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 4       words: 11       operators: 4    relational operators: 2         special char: 6         delimitors: 0
yytext: int     token: KEYWORD_INT      lineno: 5       words: 12       operators: 4    relational operators: 2         special char: 6         delimitors: 0
yytext: t       token: IDENTIFIER       lineno: 5       words: 13       operators: 4    relational operators: 2         special char: 6         delimitors: 0
yytext: =       token: ASSIGNMENT       lineno: 5       words: 13       operators: 5    relational operators: 2         special char: 6         delimitors: 0
yytext: 61      token: ICONST   lineno: 5       words: 14       operators: 5    relational operators: 2         special char: 6         delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 5       words: 14       operators: 5    relational operators: 2         special char: 7         delimitors: 0
yytext: if      token: KEYWORD_IF       lineno: 6       words: 15       operators: 5    relational operators: 2         special char: 7         delimitors: 0
yytext: (       token: LEFT_PARENTHESIS lineno: 6       words: 15       operators: 5    relational operators: 2         special char: 8         delimitors: 0
yytext: t       token: IDENTIFIER       lineno: 6       words: 16       operators: 5    relational operators: 2         special char: 8         delimitors: 0
yytext: ==      token: EQUAL_OPERATION  lineno: 6       words: 16       operators: 6    relational operators: 3         special char: 8         delimitors: 0
yytext: 61      token: ICONST   lineno: 6       words: 17       operators: 6    relational operators: 3         special char: 8         delimitors: 0
yytext: )       token: RIGHT_PARENTHESIS        lineno: 6       words: 17       operators: 6    relational operators: 3         special char: 9         delimitors: 0
yytext: {       token: LEFT_BRACE       lineno: 6       words: 17       operators: 6    relational operators: 3         special char: 10        delimitors: 0
yytext: printf  token: IDENTIFIER       lineno: 7       words: 18       operators: 6    relational operators: 3         special char: 10        delimitors: 0
yytext: (       token: LEFT_PARENTHESIS lineno: 7       words: 18       operators: 6    relational operators: 3         special char: 11        delimitors: 0
yytext: "correct number\n"      token: STRING   lineno: 7       words: 19       operators: 6    relational operators: 3         special char: 11        delimitors: 0
yytext: )       token: RIGHT_PARENTHESIS        lineno: 7       words: 19       operators: 6    relational operators: 3         special char: 12        delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 7       words: 19       operators: 6    relational operators: 3         special char: 13        delimitors: 0
yytext: }       token: RIGHT_BRACE      lineno: 8       words: 19       operators: 6    relational operators: 3         special char: 14        delimitors: 0
yytext: printf  token: IDENTIFIER       lineno: 9       words: 20       operators: 6    relational operators: 3         special char: 14        delimitors: 0
yytext: (       token: LEFT_PARENTHESIS lineno: 9       words: 20       operators: 6    relational operators: 3         special char: 15        delimitors: 0
yytext: "Hello World\n" token: STRING   lineno: 9       words: 21       operators: 6    relational operators: 3         special char: 15        delimitors: 0
yytext: )       token: RIGHT_PARENTHESIS        lineno: 9       words: 21       operators: 6    relational operators: 3         special char: 16        delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 9       words: 21       operators: 6    relational operators: 3         special char: 17        delimitors: 0
yytext: return  token: KEYWORD_RETURN   lineno: 10      words: 22       operators: 6    relational operators: 3         special char: 17        delimitors: 0
yytext: 0       token: ICONST   lineno: 10      words: 23       operators: 6    relational operators: 3         special char: 17        delimitors: 0
yytext: ;       token: SEMI_COLON       lineno: 10      words: 23       operators: 6    relational operators: 3         special char: 18        delimitors: 0
yytext: }       token: RIGHT_BRACE      lineno: 11      words: 23       operators: 6    relational operators: 3         special char: 19        delimitors: 0
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: [a+b-c]


Valid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ ./task2.l
Enter the arithmetic expression: ((([a-m+n}}


Invalid Expression
lab@PC:~/Desktop/compilor_211220061/lab4_flex$ 
```