Activities Text Editor Dec 17 1:15 PM

Open input.txt ~/Desktop/Jacob Save

lexicalanalyser.c | input.txt | output.txt

```
#include<stdio.h>
void main()
{
printf("Hello World");
}
```

Activities Text Editor Dec 17 1:15 PM

Open output.txt ~/Desktop/Jacob Save

lexicalanalyser.c | input.txt | output.txt

Line no.	Token no.	Token	Lexeme
1	0	Identifier	include
1	1	Identifier	stdio
1	2	Identifier	h
2	3	Identifier	void
2	4	Keyword	main
2	5	Specialsymbol	)
3	6	Specialsymbol	{
4	7	Identifier	printf
4	8	Identifier	Hello
4	9	Identifier	World
4	10	Specialsymbol	)
4	11	Specialsymbol	;
5	12	Specialsymbol	}

Plain Text Tab Width: 8 Ln 1, Col 1 INS

Activities Terminal Dec 17 1:32 PM

Open b.c ~/Desktop/Jacob/count_characters Save

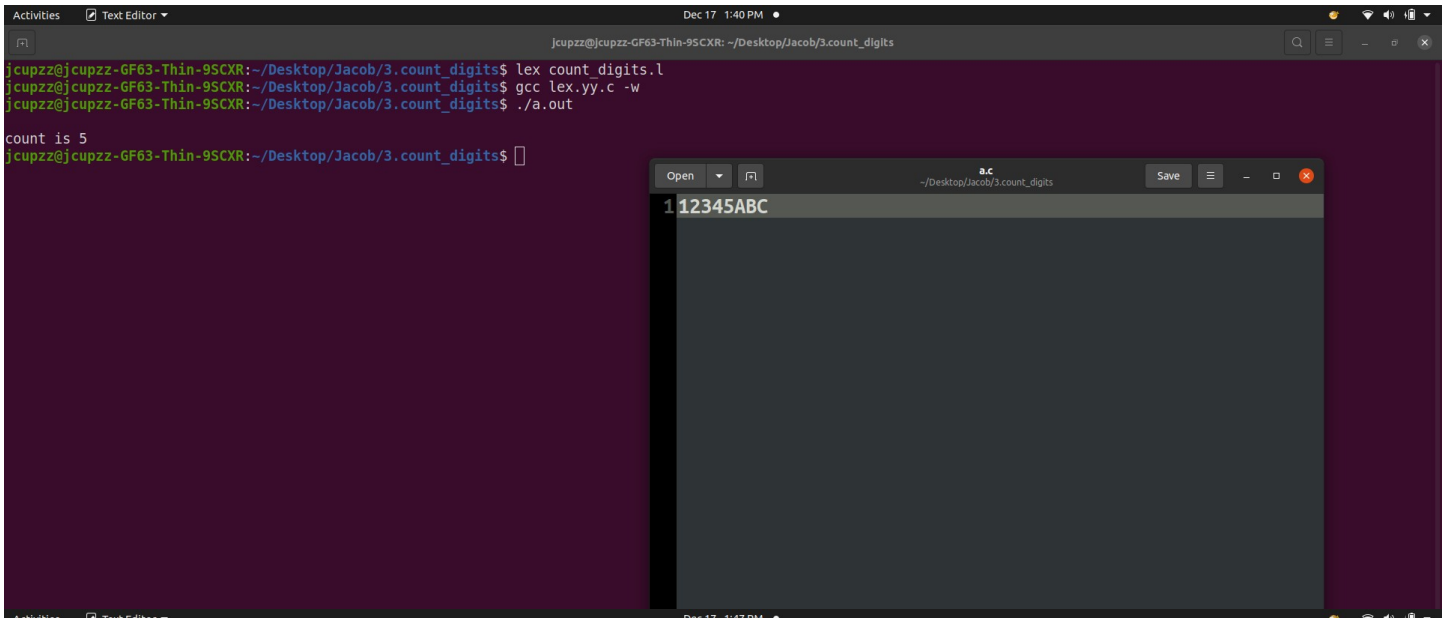
count_characters.l | b.c

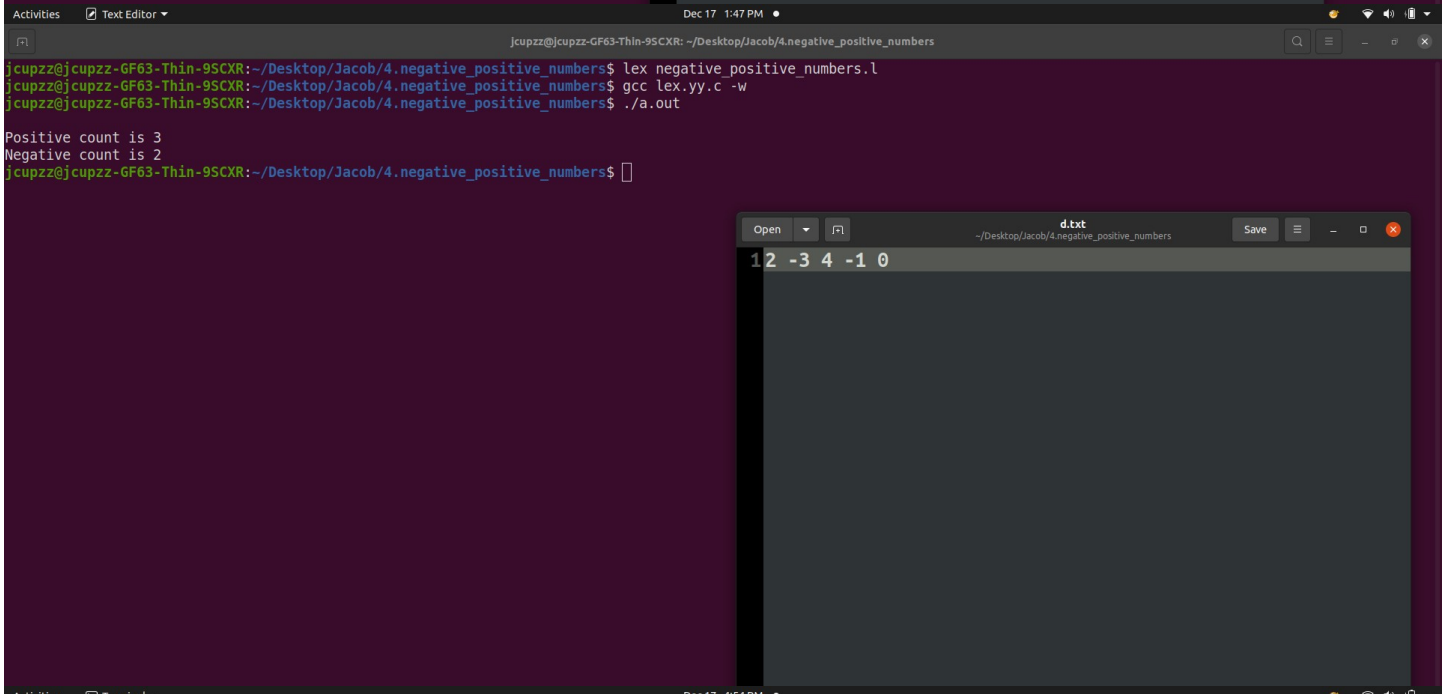
```
Jacob
```

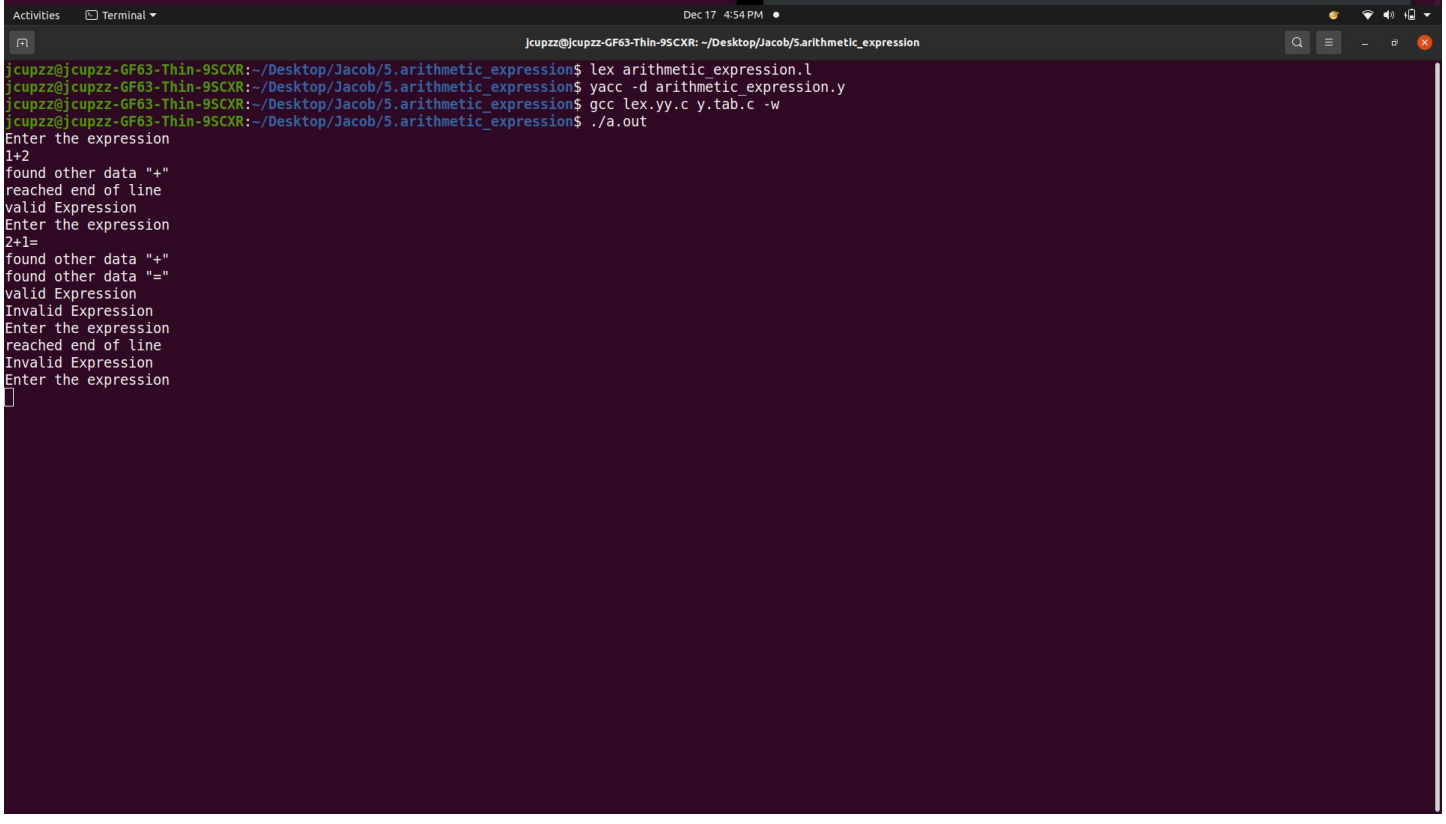
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/count_characters

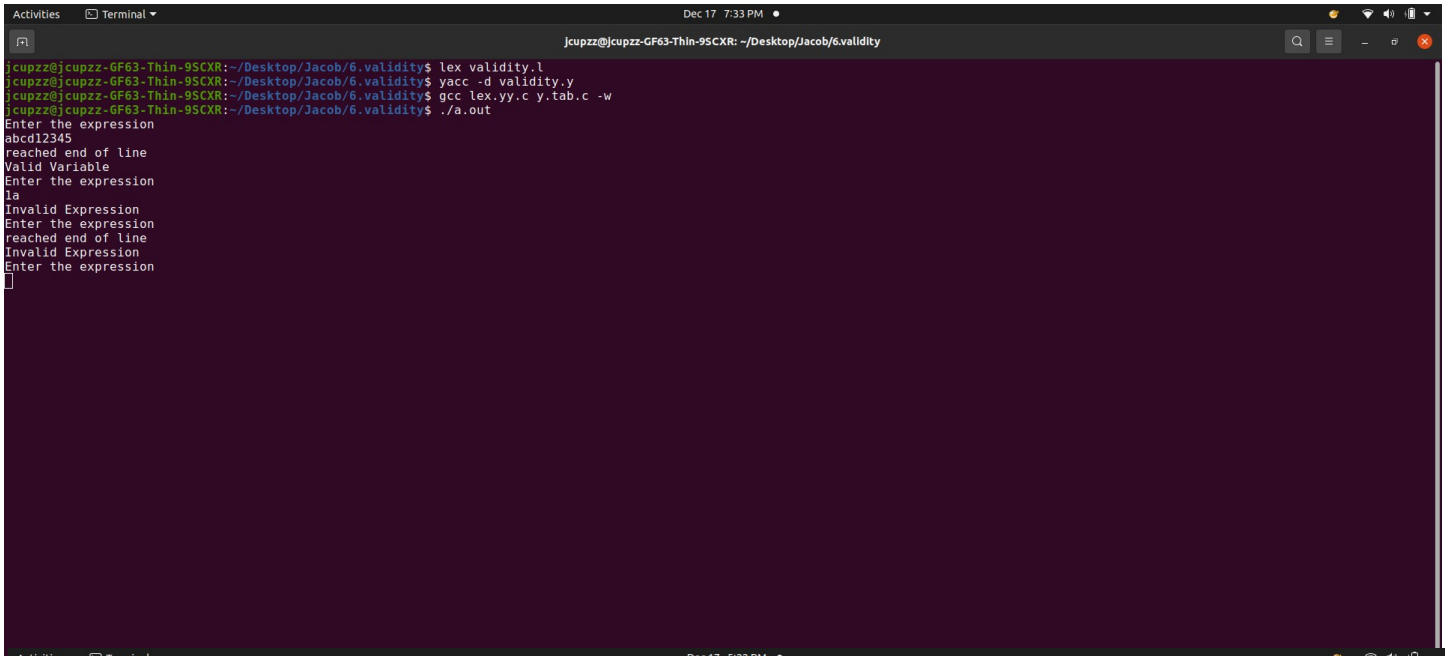
```
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/count_characters$ lex count_characters.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/count_characters$ gcc lex.yy.c
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/count_characters$ ./a.out

count is 5
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/count_characters$
```

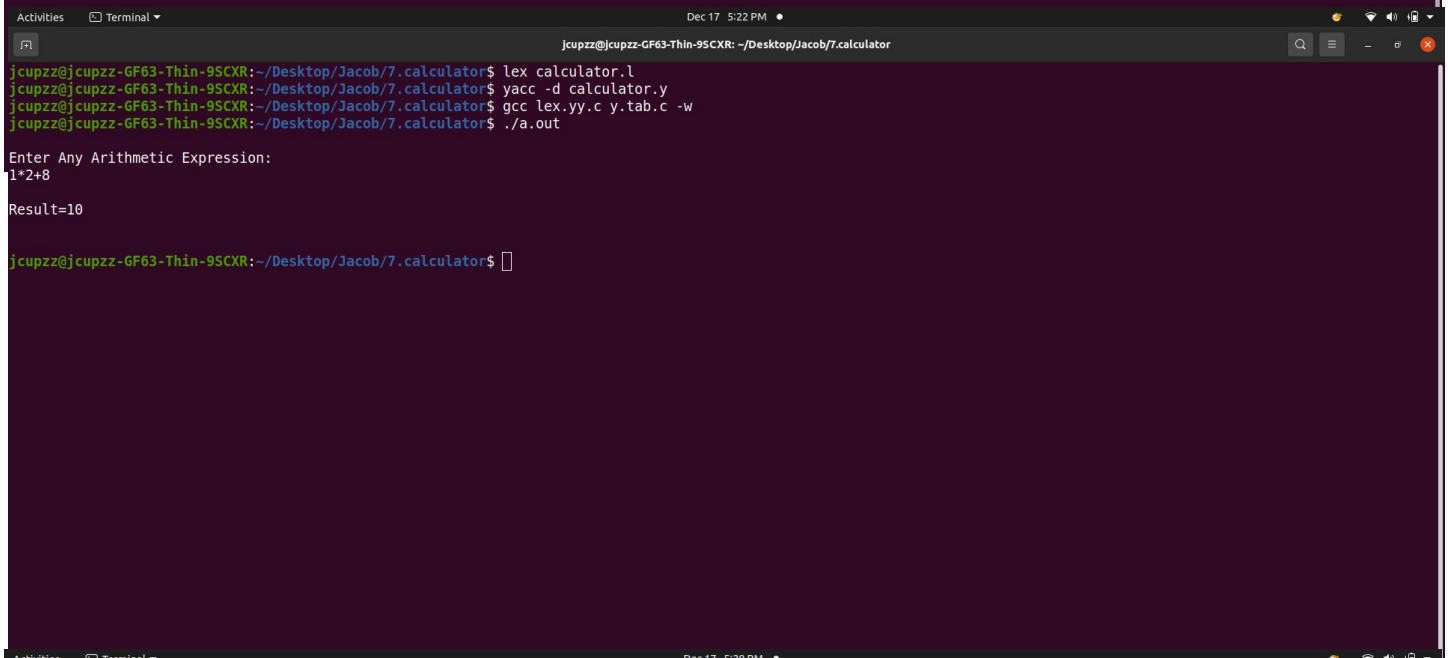
C Tab Width: 8 Ln 1, Col 6 INS

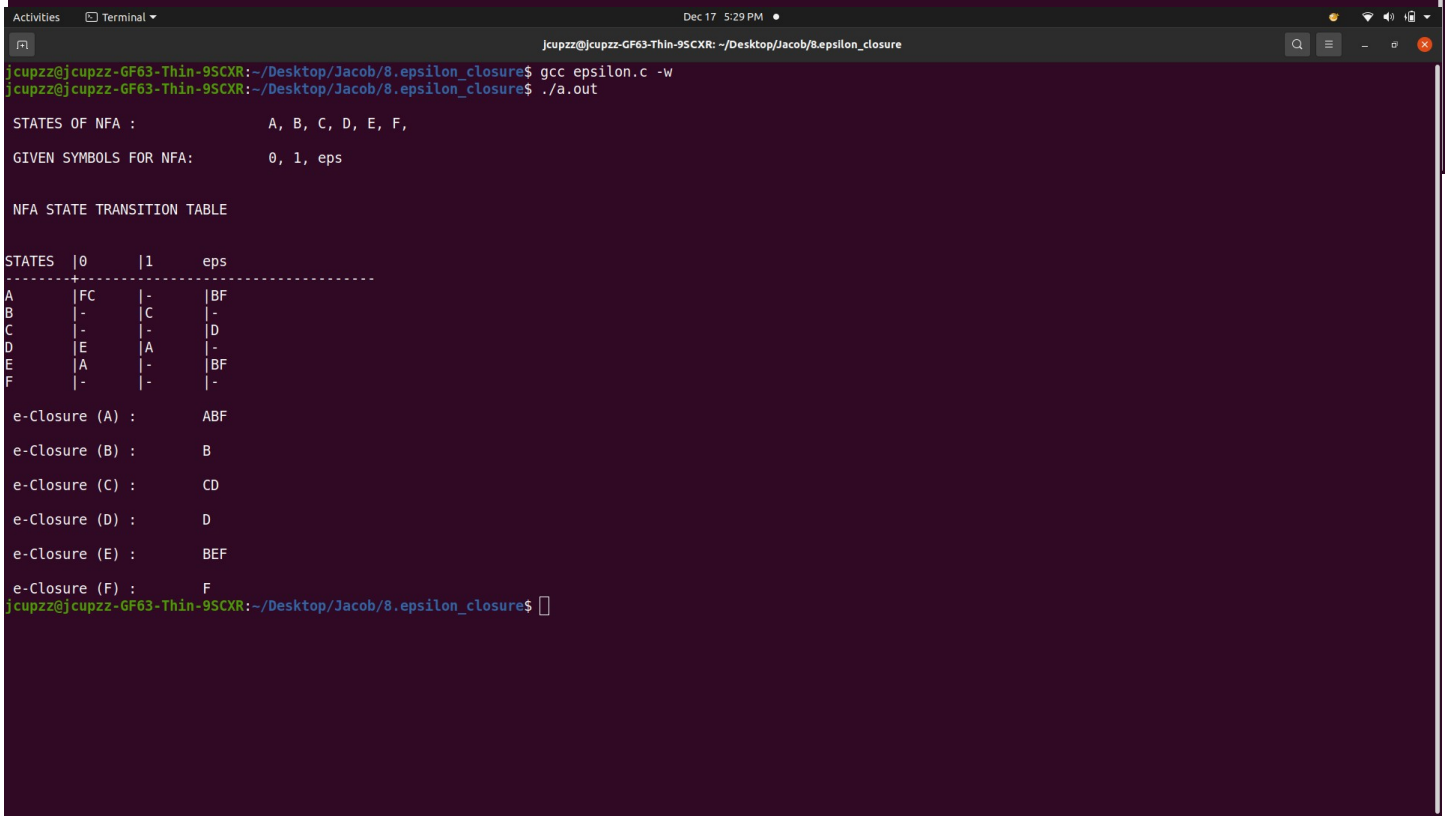

Activities
Text Editor
Dec 17 1:40 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/3.count_digits
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/3.count_digits$ lex count_digits.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/3.count_digits$ gcc lex.yy.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/3.count_digits$ ./a.out
count is 5
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/3.count_digits$
Open
a.c
~/Desktop/Jacob/3.count_digits
Save
1 12345ABC


Activities
Text Editor
Dec 17 1:47 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/4.negative_positive_numbers
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/4.negative_positive_numbers$ lex negative_positive_numbers.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/4.negative_positive_numbers$ gcc lex.yy.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/4.negative_positive_numbers$ ./a.out
Positive count is 3
Negative count is 2
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/4.negative_positive_numbers$
Open
d.txt
~/Desktop/Jacob/4.negative_positive_numbers
Save
1 2 -3 4 -1 0


Activities
Terminal
Dec 17 4:54 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/5.arithmetic_expression
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/5.arithmetic_expression$ lex arithmetic_expression.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/5.arithmetic_expression$ yacc -d arithmetic_expression.y
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/5.arithmetic_expression$ gcc lex.yy.c y.tab.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/5.arithmetic_expression$ ./a.out
Enter the expression
1+2
found other data "+"
reached end of line
valid Expression
Enter the expression
2+1=
found other data "+"
found other data "="
valid Expression
Invalid Expression
Enter the expression
reached end of line
Invalid Expression
Enter the expression

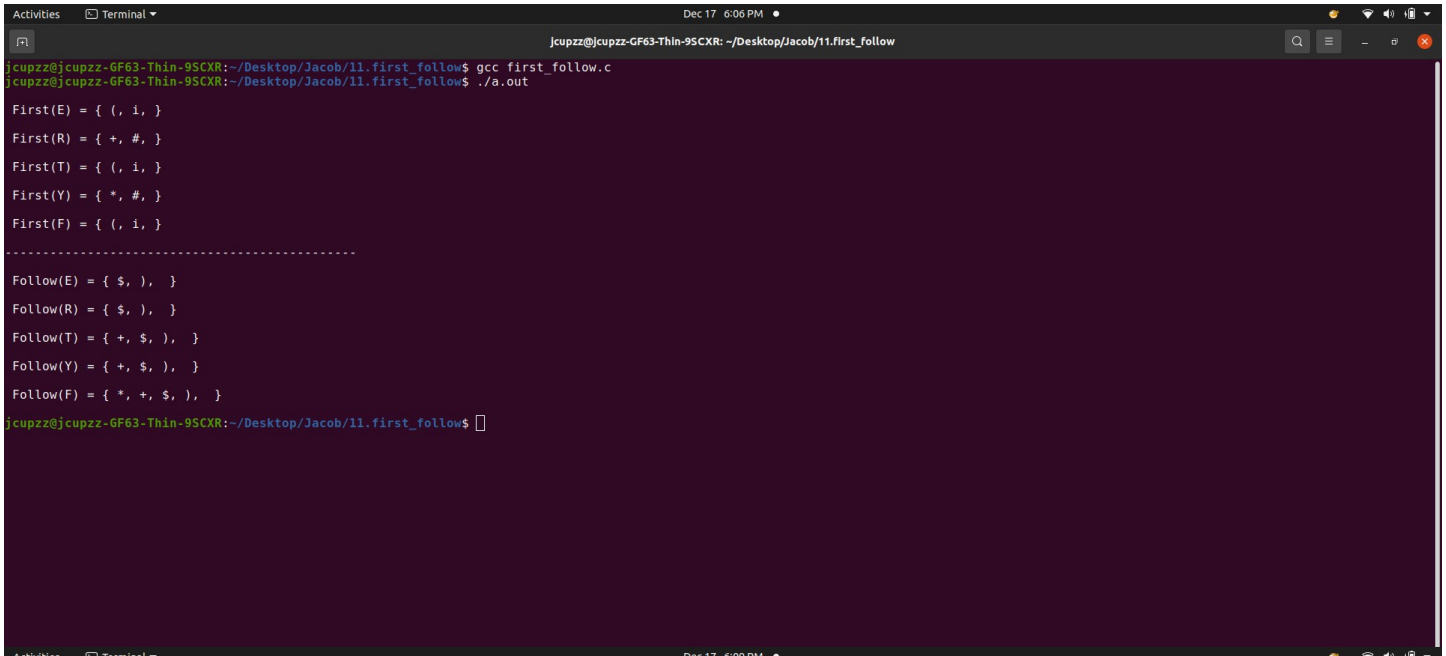

Activities
Terminal
Dec 17 7:33 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/6.validity
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/6.validity$ lex validity.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/6.validity$ yacc -d validity.y
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/6.validity$ gcc lex.yy.c y.tab.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/6.validity$ ./a.out
Enter the expression
abcd12345
reached end of line
Valid Variable
Enter the expression
1a
Invalid Expression
Enter the expression
reached end of line
Invalid Expression
Enter the expression

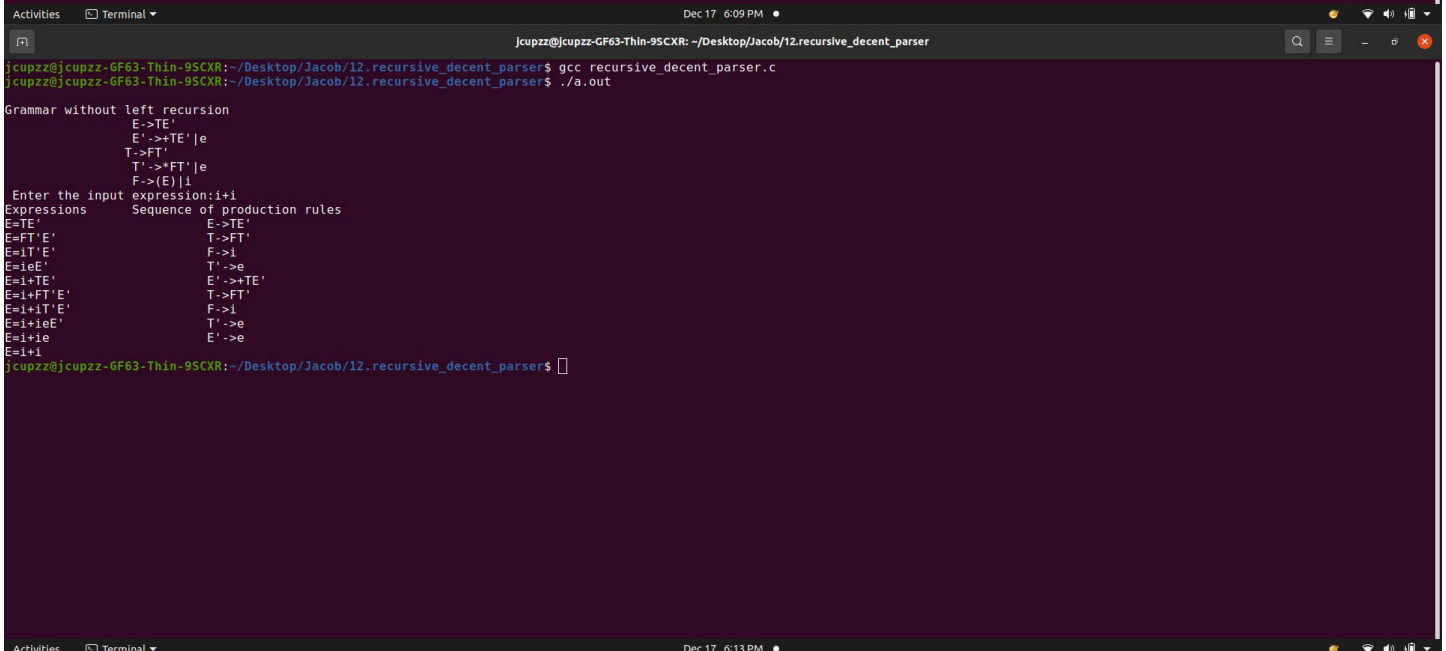

Activities
Terminal
Dec 17 5:22 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/7.calculator
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/7.calculator$ lex calculator.l
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/7.calculator$ yacc -d calculator.y
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/7.calculator$ gcc lex.yy.c y.tab.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/7.calculator$ ./a.out
Enter Any Arithmetic Expression:
1*2+8
Result=10
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/7.calculator$

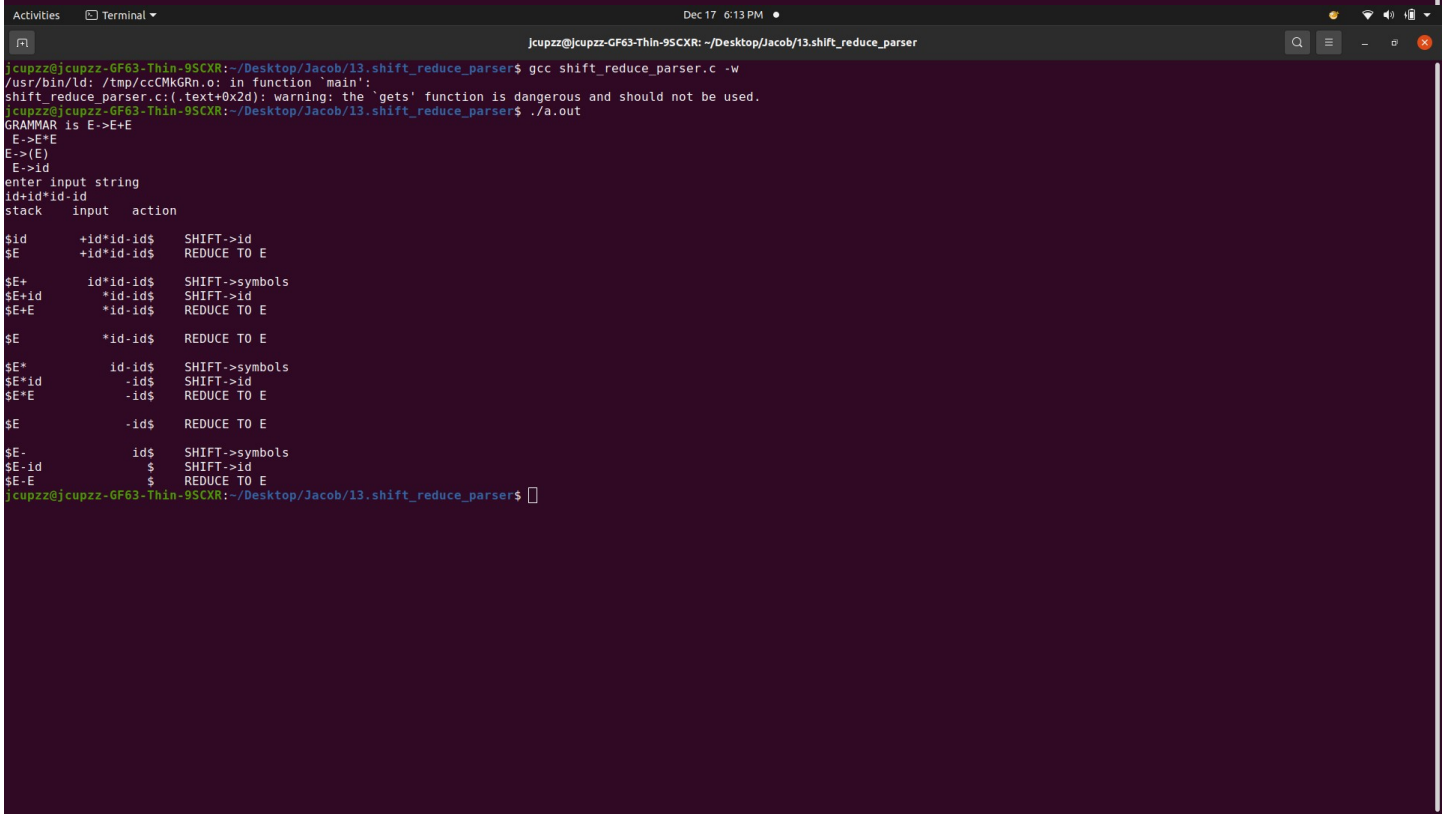

Activities
Terminal
Dec 17 5:29 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/8.epsilon_closure
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/8.epsilon_closure$ gcc epsilon.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/8.epsilon_closure$ ./a.out
STATES OF NFA : A, B, C, D, E, F,
GIVEN SYMBOLS FOR NFA: 0, 1, eps
NFA STATE TRANSITION TABLE
STATES |0 |1 eps
A |FC |- |BF
B |- |C |-
C |- |- |D
D |E |A |-
E |A |- |BF
F |- |- |-
e-Closure (A) : ABF
e-Closure (B) : B
e-Closure (C) : CD
e-Closure (D) : D
e-Closure (E) : BEF
e-Closure (F) : F
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/8.epsilon_closure$

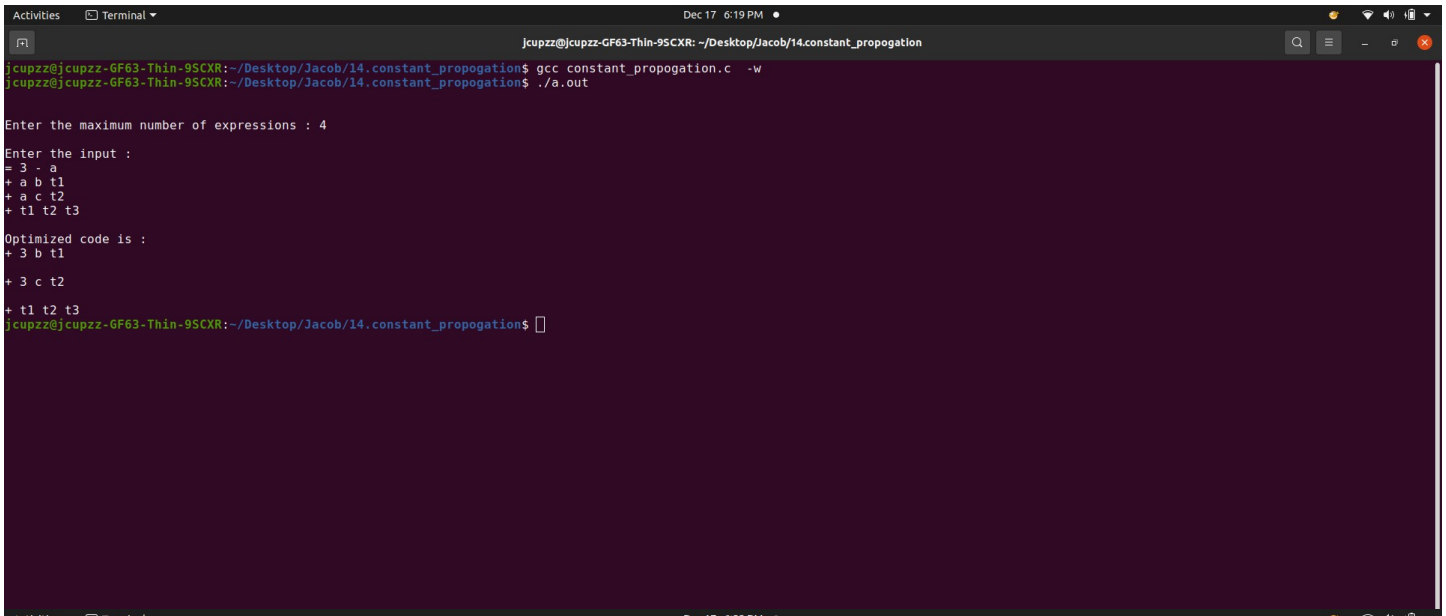
```
Activities  Terminal  Dec 17  5:57 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/9.nfa_to_dfa
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/9.nfa_to_dfa$ gcc nfa_to_dfa.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/9.nfa_to_dfa$ ./a.out

 STATES OF NFA :                A, B, C, D, E, F,

 GIVEN SYMBOLS FOR NFA:         0, 1, eps


 NFA STATE TRANSITION TABLE


STATES  |0      |1      eps
-----------------------------------------
A       |FC     |-      |BF
B       |-      |C      |-
C       |-      |-      |D
D       |E      |A      |-
E       |A      |-      |BF
F       |-      |-      |-

 e-Closure (A) :        ABF

 e-Closure (B) :        B

 e-Closure (C) :        CD

 e-Closure (D) :        D

 e-Closure (E) :        BEF

 e-Closure (F) :        F


**************************************************

                DFA TRANSITION STATE TABLE


 STATES OF DFA :                ABF, CDF, CD, BEF,

 GIVEN SYMBOLS FOR DFA:         0, 1,

STATES  |0      |1
------------------------------
ABF     |CDF    |CD
CDF     |BEF    |ABF
CD      |BEF    |ABF
BEF     |ABF    |CD
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/9.nfa_to_dfa$
```

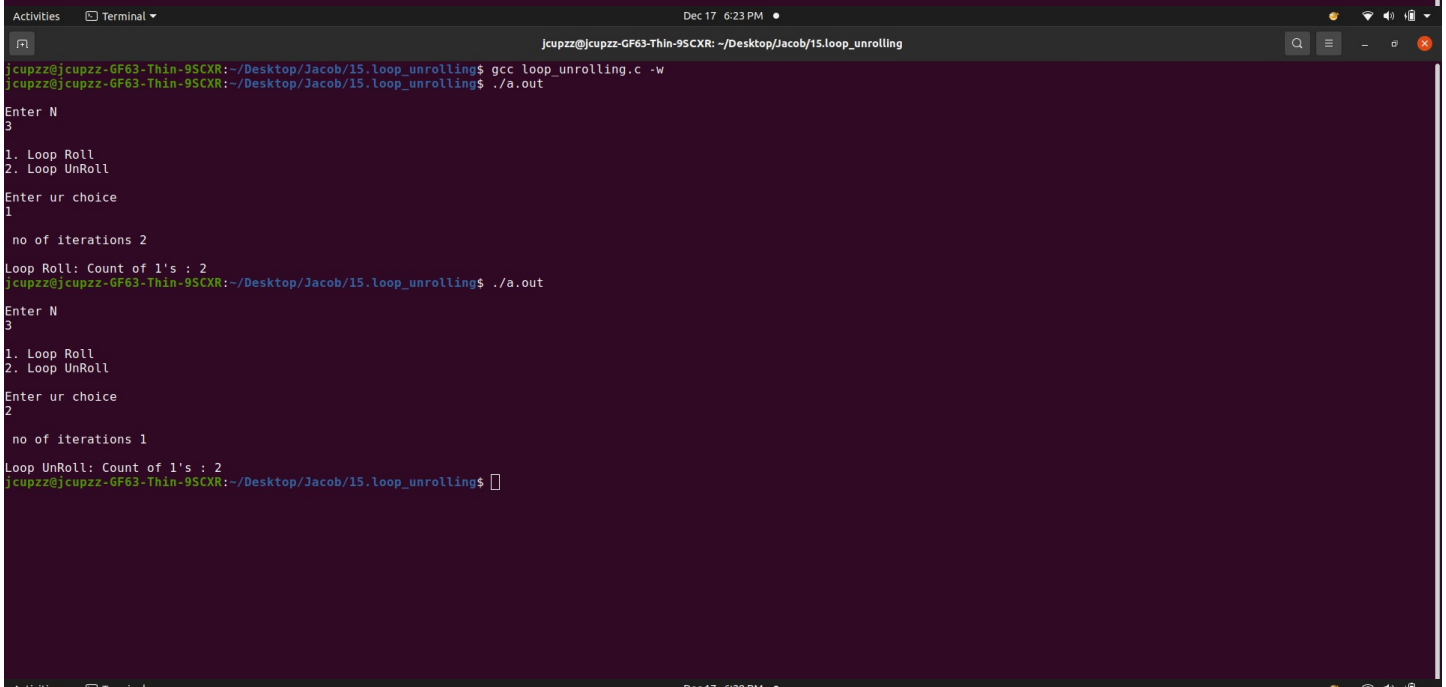
```
Activities  Terminal  Dec 17  6:03 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/10.minimize_dfa
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/10.minimize_dfa$ gcc minimize_dfa.c
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/10.minimize_dfa$ ./a.out

DFA: STATE TRANSITION TABLE
 |  0  1
-----+------------
 A |  B  C
 B |  E  F
 C |  A  A
 D |  F  E
 E |  D  F
 F |  D  E
Final states = EF

EQUIV. CLASS CANDIDATE==> 0:[ABCD] 1:[EF]
 0:[ABCD]       --> [BEAF] (0101)
 0:[ABCD]       --> [CFAE] (0101)
 1:[EF] --> [DD] (00)
 1:[EF] --> [FE] (11)

EQUIV. CLASS CANDIDATE==> 0:[AC] 1:[BD] 2:[EF]
 0:[AC] --> [BA] (10)
 0:[AC] --> [CA] (00)
 1:[BD] --> [EF] (22)
 1:[BD] --> [FE] (22)
 2:[EF] --> [DD] (11)
 2:[EF] --> [FE] (22)

EQUIV. CLASS CANDIDATE==> 0:[A] 1:[BD] 2:[C] 3:[EF]
 0:[A]  --> [B] (1)
 0:[A]  --> [C] (2)
 1:[BD] --> [EF] (33)
 1:[BD] --> [FE] (33)
 2:[C]  --> [A] (0)
 2:[C]  --> [A] (0)
 3:[EF] --> [DD] (11)
 3:[EF] --> [FE] (33)

DFA: STATE TRANSITION TABLE
 |  0  1
-----+------------
 A |  B  C
 B |  D  D
 C |  A  A
 D |  B  D
Final states = D
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/10.minimize_dfa$
```

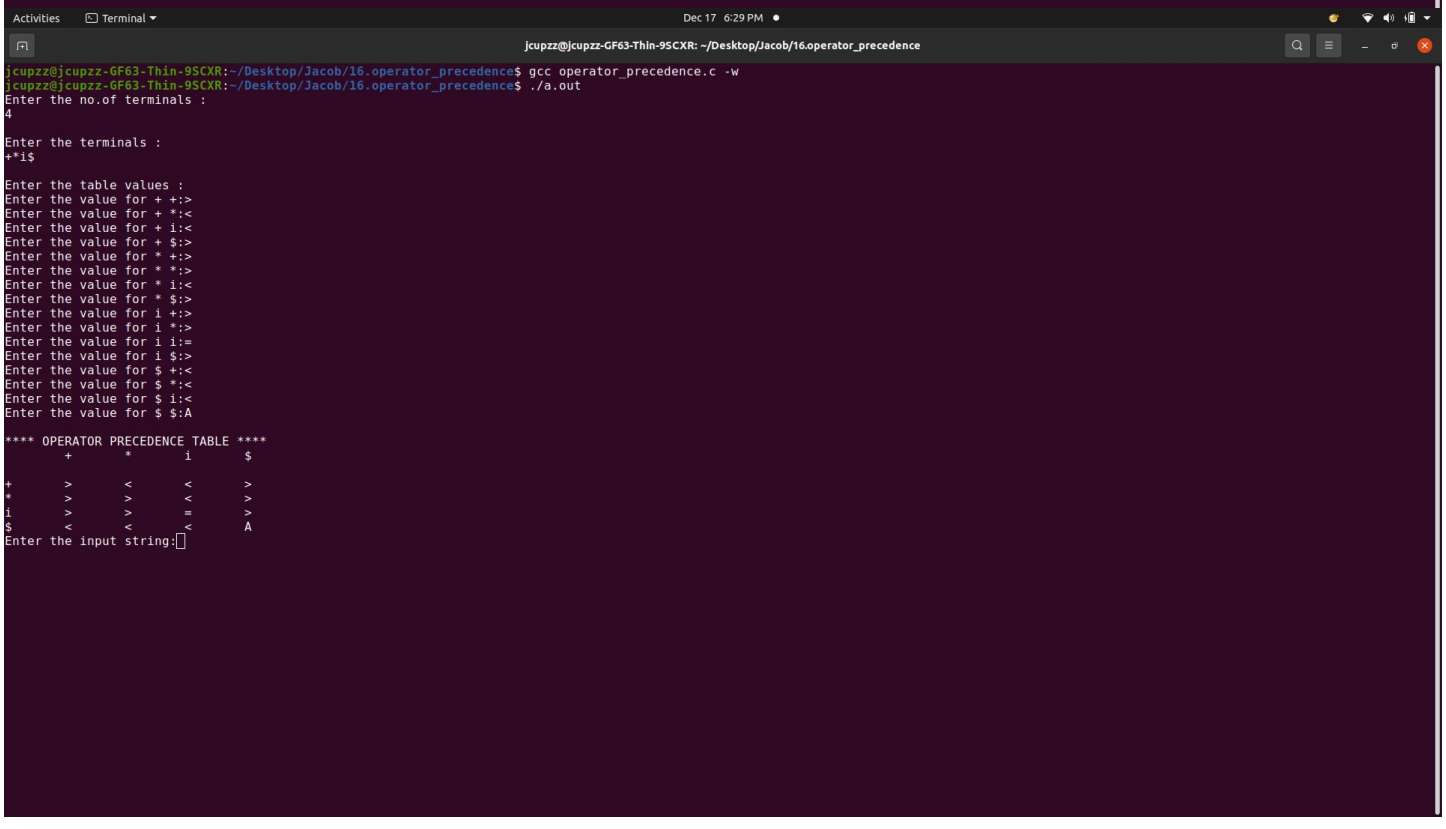

Activities
Terminal
Dec 17 6:06 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/11.first_follow
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/11.first_follow$ gcc first_follow.c
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/11.first_follow$ ./a.out
First(E) = { (, i, }
First(R) = { +, #, }
First(T) = { (, i, }
First(Y) = { *, #, }
First(F) = { (, i, }
Follow(E) = { $,), }
Follow(R) = { $,), }
Follow(T) = { +, $,), }
Follow(Y) = { +, $,), }
Follow(F) = { *, +, $,), }
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/11.first_follow$


Activities
Terminal
Dec 17 6:09 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/12.recursive_decent_parser
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/12.recursive_decent_parser$ gcc recursive_decent_parser.c
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/12.recursive_decent_parser$ ./a.out
Grammar without left recursion
E->TE'
E'->+TE'|e
T->FT'
T'->*FT'|e
F->(E)|i
Enter the input expression:i+i
Expressions Sequence of production rules
E=TE' E->TE'
E=FT'E' T->FT'
E=iT'E' F->i
E=ieE' T'->e
E=i+TE' E'->+TE'
E=i+FT'E' T->FT'
E=i+iT'E' F->i
E=i+ieE' T'->e
E=i+ie E'->e
E=i+i
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/12.recursive_decent_parser$


Activities
Terminal
Dec 17 6:13 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/13.shift_reduce_parser
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/13.shift_reduce_parser$ gcc shift_reduce_parser.c -w
/usr/bin/ld: /tmp/ccCMkGRn.o: in function `main':
shift_reduce_parser.c:(.text+0x2d): warning: the `gets' function is dangerous and should not be used.
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/13.shift_reduce_parser$ ./a.out
GRAMMAR is E->E+E
E->E*E
E->(E)
E->id
enter input string
id+id*id-id
stack input action
$id +id*id-id$ SHIFT->id
$E +id*id-id$ REDUCE TO E
$E+ id*id-id$ SHIFT->symbols
$E+id *id-id$ SHIFT->id
$E+E *id-id$ REDUCE TO E
$E *id-id$ REDUCE TO E
$E* id-id$ SHIFT->symbols
$E*id -id$ SHIFT->id
$E*E -id$ REDUCE TO E
$E -id$ REDUCE TO E
$E- id$ SHIFT->symbols
$E-id $ SHIFT->id
$E-E $ REDUCE TO E
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/13.shift_reduce_parser$


Activities
Terminal
Dec 17 6:19 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/14.constant_propogation
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/14.constant_propogation$ gcc constant_propogation.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/14.constant_propogation$ ./a.out
Enter the maximum number of expressions : 4
Enter the input :
= 3 - a
+ a b t1
+ a c t2
+ t1 t2 t3
Optimized code is :
+ 3 b t1
+ 3 c t2
+ t1 t2 t3
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/14.constant_propogation$


Activities
Terminal
Dec 17 6:23 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/15.loop_unrolling
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/15.loop_unrolling$ gcc loop_unrolling.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/15.loop_unrolling$ ./a.out
Enter N
3
1. Loop Roll
2. Loop UnRoll
Enter ur choice
1
no of iterations 2
Loop Roll: Count of 1's : 2
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/15.loop_unrolling$ ./a.out
Enter N
3
1. Loop Roll
2. Loop UnRoll
Enter ur choice
2
no of iterations 1
Loop UnRoll: Count of 1's : 2
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/15.loop_unrolling$


Activities
Terminal
Dec 17 6:29 PM
jcupzz@jcupzz-GF63-Thin-9SCXR: ~/Desktop/Jacob/16.operator_precedence
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/16.operator_precedence$ gcc operator_precedence.c -w
jcupzz@jcupzz-GF63-Thin-9SCXR:~/Desktop/Jacob/16.operator_precedence$ ./a.out
Enter the no.of terminals :
4
Enter the terminals :
+*i$
Enter the table values :
Enter the value for + +:>
Enter the value for + *:<
Enter the value for + i:<
Enter the value for + $:>
Enter the value for * +:>
Enter the value for * *:>
Enter the value for * i:<
Enter the value for * $:>
Enter the value for i +:>
Enter the value for i *:>
Enter the value for i i:=
Enter the value for i $:>
Enter the value for $ +:<
Enter the value for $ *:<
Enter the value for $ i:<
Enter the value for $ $:A
**** OPERATOR PRECEDENCE TABLE ****
+ * i $
+ > < < >
* > > < >
i > > = >
$ < < < A
Enter the input string: