

**Test Cases:**

## 1) Without error (Nested if-else)

```
srilekha@srilekha-asus:~/CS251-A2$ lex parser.l
srilekha@srilekha-asus:~/CS251-A2$ yacc parser.y
srilekha@srilekha-asus:~/CS251-A2$ gcc y.tab.c -ll -w
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-1.c
```

Parsing complete

-----Symbol Table-----

Token	Token Type
-------	------------

a	INT
main	PROCEDURE

-----CONSTANT TABLE-----

Value	Data Type
-------	-----------

4	INT
15	INT
1	INT
2	INT

```
srilekha@srilekha-asus:~/CS251-A2$
```

## 2) Without error (While and for loop)

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-2.c
```

Parsing complete

-----Symbol Table-----

Token	Token Type
-------	------------

a	INT
i	INT
main	PROCEDURE

-----CONSTANT TABLE-----

Value	Data Type
-------	-----------

4	INT
15	INT
1	INT
5	INT

```
srilekha@srilekha-asus:~/CS251-A2$
```

3) With error (Missing multiple semicolon)

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-3.c
Line 6 : parse error
Parsing failed

-----Symbol Table-----
Token      |      Token Type
-----
b           |      INT
main        |      PROCEDURE
printf      |      PROCEDURE
-----

-----CONSTANT TABLE-----
Value      |      Data Type
-----
5           |      INT

srilekha@srilekha-asus:~/CS251-A2$
```

4) With error (Misspelled keyword 'while' and unmatched parentheses)

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-4.c
Line 6 : parse error
Parsing failed

-----Symbol Table-----
Token      |      Token Type
-----
a           |      INT
main        |      PROCEDURE
-----

-----CONSTANT TABLE-----
Value      |      Data Type
-----
4           |      INT
15          |      INT

srilekha@srilekha-asus:~/CS251-A2$
```

5) With error (Dangling else problem)

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-5.c

Line 13 : parse error
Parsing failed

-----Symbol Table-----
Token          | Token Type
-----
a               | INT
main            | PROCEDURE
printf          | PROCEDURE
-----

-----CONSTANT TABLE-----
Value          | Data Type
-----
4              | INT
15             | INT
"15"           | STRING
12             | INT
"CS251"        | STRING
"CS252"        | STRING

srilekha@srilekha-asus:~/CS251-A2$
```

6) With error (for-loop syntax error)

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out testcase-6.c

Line 4 : parse error
Parsing failed

-----Symbol Table-----
Token          | Token Type
-----
a               | INT
i               | INT
main            | PROCEDURE
-----

-----CONSTANT TABLE-----
Value          | Data Type
-----
4              | INT
0              | INT
15             | INT

srilekha@srilekha-asus:~/CS251-A2$
```

## Executing provided Samples from Canvas:

```
srilekha@srilekha-asus:~/CS251-A2$ lex parser.l
srilekha@srilekha-asus:~/CS251-A2$ yacc parser.y
srilekha@srilekha-asus:~/CS251-A2$ gcc y.tab.c -ll -w
srilekha@srilekha-asus:~/CS251-A2$ ./a.out test11.c
```

Line 0 : parse error

Parsing failed

```
-----Symbol Table-----
-----
Token          |      Token Type
-----
```

```
-----CONSTANT TABLE-----
-----
Value          |      Data Type
-----
```

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out test22.c
```

Line 0 : parse error

Parsing failed

```
-----Symbol Table-----
-----
Token          |      Token Type
-----
```

```
-----CONSTANT TABLE-----
-----
Value          |      Data Type
-----
```

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out test33.c
```

```
Line 0 : parse error
```

```
Parsing failed
```

```
-----Symbol Table-----  
-----  
Token          |      Token Type  
-----  
-----
```

```
-----CONSTANT TABLE-----  
-----  
Value          |      Data Type  
-----  
-----
```

```
srilekha@srilekha-asus:~/CS251-A2$ ./a.out test44.c
```

```
Line 0 : parse error
```

```
Parsing failed
```

```
-----Symbol Table-----  
-----  
Token          |      Token Type  
-----  
-----
```

```
-----CONSTANT TABLE-----  
-----  
Value          |      Data Type  
-----  
-----
```

```
srilekha@srilekha-asus:~/CS251-A2$
```