

```
1 Test case 1
2
3 Enter the string to parsed :a+b*(a+b)
4 Enter filename for the set of production
5 test.txt
6
7 Input file test.txt:
8
9 E->E+E
10 E->a
11 E->E*E
12 E->(E)
13 E->b
14
15 Output:
16
17 -----BEGIN PARSING STEPS-----
18
19     Shift a into stack
20     Reduce by E->a
21     Shift + into stack
22     Shift b into stack
23     Reduce by E->b
24     Reduce by E->E+E
25     Shift * into stack
26     Shift ( into stack
27     Shift a into stack
28     Reduce by E->a
29     Shift + into stack
30     Shift b into stack
31     Reduce by E->b
32     Reduce by E->E+E
33     Shift ) into stack
34     Reduce by E->(E)
35     Reduce by E->E*E
36
37 -----String Parsed Successfully-----
38
39 Test case 2
40
41 Enter the string to parsed :a+b*(a/c)
42 Enter filename for the set of production
43 test.txt
44
45 Input file test.txt:
46
47 E->E+E
48 E->a
49 E->E*E
50 E->(E)
51 E->b
52
53 -----BEGIN PARSING STEPS-----
54
55     Shift a into stack
56     Reduce by E->a
57     Shift + into stack
```

```
58      Shift b into stack
59      Reduce by E->b
60      Reduce by E->E+E
61      Shift * into stack
62      Shift ( into stack
63      Shift a into stack
64      Reduce by E->a
65      Shift / into stack
66      Shift c into stack
67      Shift ) into stack
```

68

69

70 The string does not belong the the grammar...

71

72

73

74