

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <ctype.h>
4  char input[100];
5  int i = 0;
6  int E();
7  int EP();
8  int T();
9  int TP();
10 int F();
11 int main() {
12     int choice;
13     printf("\nRecursive descent parsing for the following grammar\n");
14     printf("E  -> T E'\n");
15     printf("E' -> + T E' / ε\n");
16     printf("T  -> F T'\n");
17     printf("T' -> * F T' / ε\n");
18     printf("F  -> (E) / id\n");
19     do {
20         i = 0;
21         printf("\nEnter the string to be checked: ");
22         fgets(input, sizeof(input), stdin);
23         input[strcspn(input, "\n")] = '\0';
24         if (E() && input[i] == '\0')
25             printf("\n✅ String is accepted\n");
26         else
27             printf("\n❌ String is not accepted\n");
28         printf("\nDo you want to test another string? (1=Yes / 0=No): ");
29         scanf("%d", &choice);
30         getchar();
31     } while (choice == 1);
32
33     return 0;
34 }
```

```
35 int E() {
36     if (T()) {
37         if (EP()) return 1;
38     }
39     return 0;
40 }
41 int EP() {
42     if (input[i] == '+') {
43         i++;
44         if (T() && EP()) return 1;
45         return 0;
46     }
47     return 1;
48 }
49 int T() {
50     if (F()) {
51         if (TP()) return 1;
52     }
53     return 0;
54 }
55 int TP() {
56     if (input[i] == '*') {
57         i++;
58         if (F() && TP()) return 1;
59         return 0;
60     }
61     return 1;
62 }
```

```
63 int F() {  
64     if (input[i] == '(') {  
65         i++;  
66         if (E() && input[i] == ')') {  
67             i++;  
68             return 1;  
69         }  
70         return 0;  
71     } else if (isalpha(input[i])) {  
72         i++;  
73         return 1;  
74     }  
75     return 0;  
76 }
```

Recursive descent parsing for the following grammar

$E \rightarrow T E'$


$E' \rightarrow + T E' / \epsilon$

$T \rightarrow F T'$

$T' \rightarrow * F T' / \epsilon$

$F \rightarrow (E) / id$

Enter the string to be checked: (a+b)\*c

 String is accepted

Do you want to test another string? (1=Yes / 0=No): 1

Enter the string to be checked: a/c+d

 String is not accepted

Do you want to test another string? (1=Yes / 0=No): 0

=== Code Execution Successful ===