2) Given a balanced parenthesis string s, return the score of that string

ans)

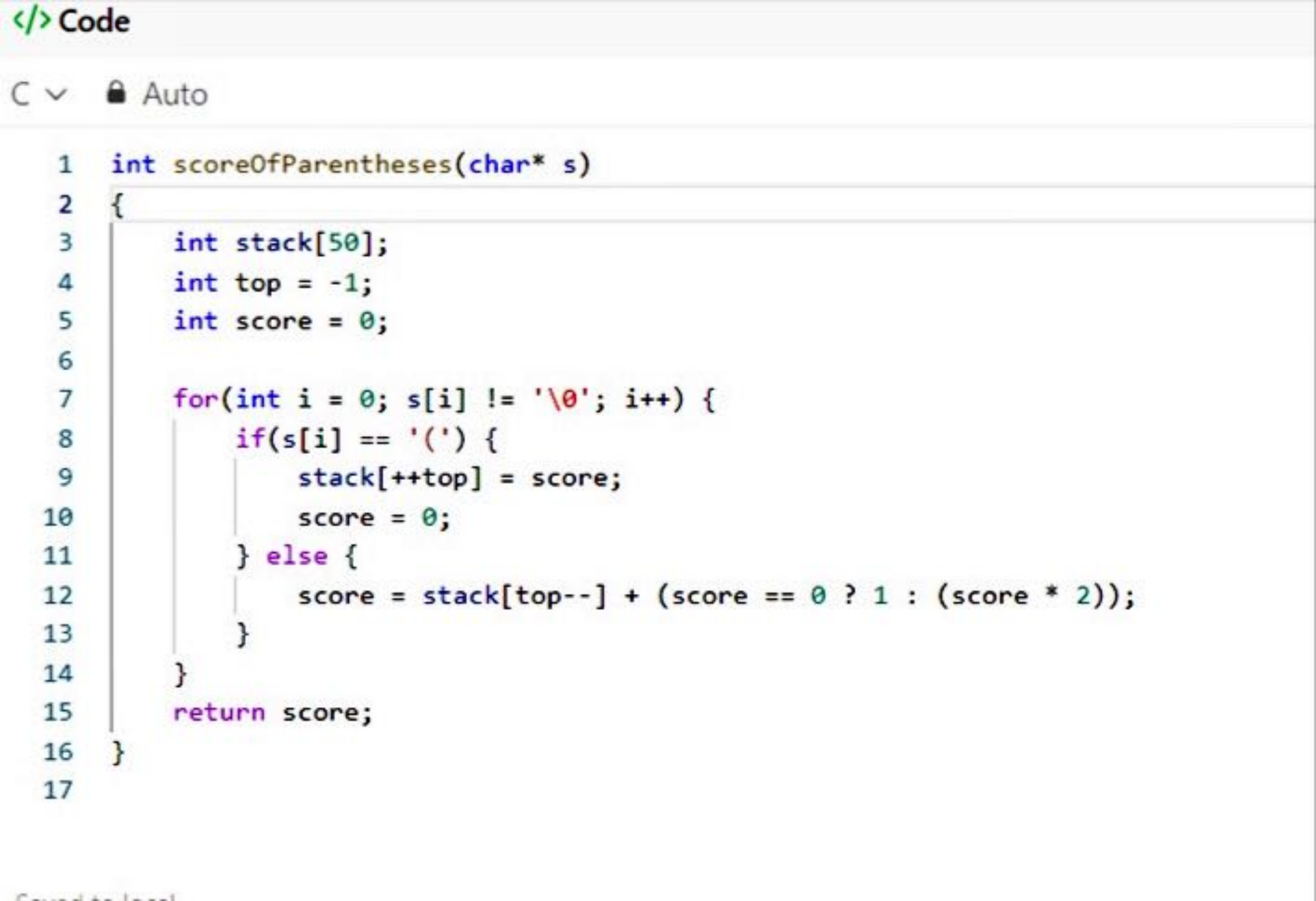
```
#include<stdio.h>
# include <stdlib.a>
int  scoreofparantheis (char * s)
{
      int  stack [50];
      int  top =-1, score =0;
      for (int  i =0; s[i] !='\0'; i++)
       {
            if (s[i] == '(')
            {
                 stack [++top] = score;
                 score =0;
            }

            else
            {
```

19/2/24

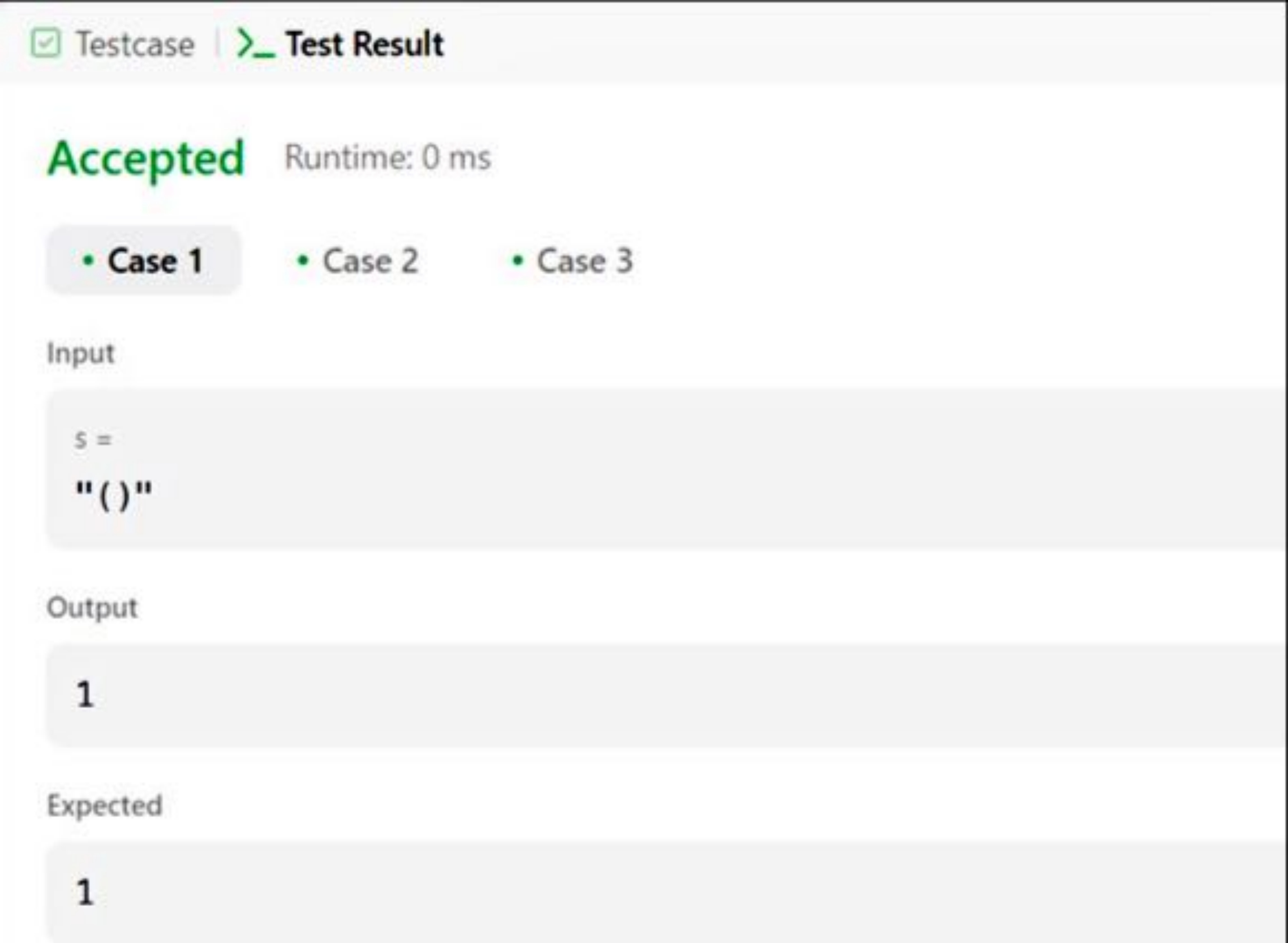
```
            score = stack [top --] + score = = 0 ? 1 : (score * 2);
        }
      }
      return  score;
}
void  main()
{
      char  s[50];
      printf (" Enter the  string : ");
      scanf (" %s" , s);
      printf (" \n  score = %d", scoreofparanthesis (s));
}
```

OUTPUT :

```
Enter  the  string : (())
score = 2
```



```c
int scoreOfParentheses(char* s)
{
    int stack[50];
    int top = -1;
    int score = 0;

    for(int i = 0; s[i] != '\0'; i++) {
        if(s[i] == '(') {
            stack[++top] = score;
            score = 0;
        } else {
            score = stack[top--] + (score == 0 ? 1 : (score * 2));
        }
    }
    return score;
}
```



Accepted Runtime: 0 ms

Input

```
s =
"()"
```

Output

```
1
```

Expected

```
1
```