Assignment 1: Stack

~ 9/30 11:59 PM

Notification

The task should be done by yourself, and you can't use codes from Internet or anyone else. If you don't follow this rule, we will give you 0 score and there can be other disadvantages like F grade.

Exam Guide

1. **This test will be conducted in Groom and scoring results will not be released until 9/30.**
2. **A perfect score is 100 points**
3. **Each question will be scored with multiple test cases and scored based on the number of passed test cases.**
4. **Please keep the submission deadline.**
5. **Please read notification about assignment on I-campus before start your assignment.**

Problem Lists

Problem 1. 40 pts

Problem 2. 60 pts

Problem 1

Mountain

Score: 40pts

There is a string with two characters: '/' , '\'. We will use that string to create a 'mountain'. '/' means mountain's uphill '\' means mountain's downhill. Check that when you put the letters in the string in order, you can create a normal mountain. A normal mountain should start with'/'. And if there is a'/', there is a corresponding'\'. Also, if there is a'\', there is a corresponding '/'.

Examples of normal mountains:

/\\	//\\	///\\/\/\\	/\/\/\\
/\\	/\ / \	/\ / \/\/\ / \	/\/\ /\/ \

Examples of wrong mountains:

\/	/	///\\	\/\/\/\\	/\/\\/\\
\/	/	/\ / \ /	/\ \/\/ \	/\/\ /\ \/ \

- **Input**
  - Only one string without blank.
  - The length of string is N. **0 < N < 1000**

- **Output**
  - If a normal mountain is made, print the height of the mountain.
  - Output 0 if wrong mountain is made.

Sample Input 1	Sample Output 1
///\\/\/\\	3

Sample Input 2	Sample Output 2
///\\	0

Problem 2
Super Parenthesis Value
Score: 60pts

There is a string with six characters: '{' , '}' , '[' , ']' , '(' , ')'. First you have to check that the string is a normal string. Normal string follow this rule:

1. '{}', '()' and '[]', which consist only of one pair of parentheses, are correct parentheses.
2. If X is a valid parentheses column, then '{X}', '(X)' and '[X]' are also valid parentheses.
3. If both X and Y are correct parentheses, then XY that combines them will also be correct parentheses.

For example, '(){([[]])}' and '(())[][]{}' is a valid parentheses column, but '([)]' and '(){)[]}]' are not valid parentheses. We define the value of the parenthesis (the parenthesis value) as follows for any correct parentheses X and denote it as the value (X).

1. The value of the parenthesis of '{}' is -1.
2. The value of the parenthesis of '()' is 2.
3. The value of the parenthesis of '[]' is 3.
4. The parenthesis value of '{X}' is calculated as '-1 × value (X)'.
5. The parenthesis value of '(X)' is calculated as '2 × value(X)'.
6. The parenthesis value of '[X]' is calculated as '3 × value (X)'.
7. The parenthesis value of XY where the correct parentheses X and Y are combined is calculated as value (XY) = value (X) + value (Y).

For example, let's find the parenthesis value of '{(()[[]])([])}'. The parenthesis value of '()[[]]' is 2 + 3×3=11, the parenthesis value of '(()[[ ]])' is 2×11=22. And the value of '([])' is 2×3=6, so the value of this parenthesis column is 22 + 6 = 28. Since the parentheses are enclosed in the outermost {}, total parenthesis value is multiplying by -1 then becomes -28.

➢ Input

- Only one string without blank.
- The length of string is N. **0 < N < 30**

➢ Output

- Print the final parenthesis value on the first line.
- If the input is an invalid parenthesis string, 0 must be output.

Sample Input 1	Sample Output 1
{(()[[]])([])}	-28

Sample Input 2	Sample Output 2
{[}]	0

- This is the last page. Good luck ☺ -