



9

### Galactic Bracket Balance Equilibrium

4 points

#### Introduction

In a distant future, humanity has colonized numerous planets and galaxies. With advanced technology and interstellar travel, the cosmos has become the playground of explorers, adventurers, and mathematicians alike.

One day, while navigating through a wormhole, your spaceship encounters a peculiar cosmic phenomenon: a rift in space-time that distorts reality itself. Your ship's computer, equipped with advanced AI, detects a mysterious signal emanating from this rift. Upon closer inspection, you realize it's not a signal but a sequence of brackets - opening brackets '(' and closing brackets ')' - arranged in a peculiar pattern.

As an intrepid astronaut-mathematician, you are tasked with deciphering the secrets hidden within this cosmic anomaly. Your mission is to find the "Galactic Bracket Balance Equilibrium" point within the string of brackets.

An equal point in this cosmic string is the index  $k$  where the number of opening brackets '(' occurring before index  $k$  is precisely equal to the number of closing brackets ')' occurring from index  $k$  onwards. The counting for index  $k$  will start always at 0.

Picture yourself in your spacesuit, floating near this rift in space-time, laser-scanning the brackets, and running your AI algorithms to solve this otherworldly puzzle.

Write a program that takes the cosmic string as input and returns the index of the Galactic Bracket Balance Equilibrium point.

#### Input

A cosmic string  $s$  that consists of only two characters: '(' and ')'.

#### Output

An integer representing the index of the Galactic Bracket Balance Equilibrium point followed by the balanced and split bracket sequence.





### Example 1

**Input**

((()())

**Output**

3 (()-())

### Example 2

**Input**

(( (

**Output**

0 -(( (

### Example 3

**Input**

(( ( ( ) ) ) )

**Output**

3 (( (- ) ) )

### Example 4

**Input**

( ) ( )

**Output**

3 ( ) - ( )

### Example 5

**Input**

) )

**Output**

2 )) -