

Poprawianie nawiasowania (nawiasowanie-upd)

Limit pamięci: 64 MB

Limit czasu: 0.50 s

Jasio mówi zdecydowanie zbyt dużo, zatem jego nauczyciele są często zmuszeni do wymyślania żmudnych zadań, by zająć chłopca choćby na chwilę. Tym razem, na zajęciach z parsowania, Jasiowi przyszło się zmierzyć z nawiasami. Nauczyciel wręczył mu wydruk z ciągiem symboli (oraz), który niestety niekoniecznie jest poprawnym nawiasowaniem.

Poprawne nawiasowania możemy zdefiniować rekurencyjnie w następujący sposób:

- ciąg pusty jest poprawnym nawiasowaniem,
- jeśli S i T są poprawnymi nawiasowaniami, to ich złączenie ST także jest poprawnym nawiasowaniem,
- jeśli S jest poprawnym nawiasowaniem, to (S) także.

Nauczyciel polecił, by poprawić ciąg za pomocą operacji odwracania nawiasów, to znaczy zmian pojedynczego symbolu $)$ na $($ lub odwrotnie. Jasio ma mnóstwo pytań czekających na odpowiedź, chciałby zatem szybko wyznaczyć minimalną liczbę nawiasów, które trzeba odwrócić, aby uzyskać poprawne nawiasowanie – pomóż mu.

Napisz program, który: wczyta nawiasowanie dane przez nauczyciela, wyznaczy minimalną liczbę zmian, aby uzyskać poprawne nawiasowanie i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się niepusty ciąg nawiasów (lub) o parzystej długości.

Wyjście

W pierwszym i jedynym wierszu wyjścia powinna się znaleźć jedna liczba całkowita – minimalna liczba nawiasów, które należy odwrócić, aby uzyskać poprawne nawiasowanie.

Ograniczenia

Długość ciągu z wejścia nie przekracza miliona znaków.

Przykład

Wejście

)((())

Wyjście

2

Wyjaśnienie

Wystarczy odwrócić nawiasy pierwszy i drugi.

Wejście

()))(((

Wyjście

3