



Zadanie: MRO

Mrówki

Potyczki Algoritmiczne 2024, runda próbna. Limity: 1024 MB, 1 s.

05.03.2024

Na osi liczbowej stoi n mrówek – i -ta z nich w punkcie i . Każda z mrówek patrzy w prawo (w kierunku rosnących współrzędnych) lub w lewo (w kierunku malejących współrzędnych). Mrówki są na tyle małe, że możemy traktować je jak pojedyncze punkty.

Na sygnał wszystkie mrówki zaczynają z jednakową, jednostkową prędkością maszerować w kierunkach w które patrzą. Jeśli dwie mrówki się zderzą (znajdą się w tym samym punkcie), to odbijają się od siebie, tzn. obie zmieniają kierunek marszu i maszerują dalej. Można udowodnić, że po pewnym czasie nie będą już więcej następować żadne zderzenia. Czy jesteś w stanie napisać program, który dla każdej mrówki obliczy ile razy odbije się od innych mrówek?

Wejście

W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita n ($1 \leq n \leq 300\,000$), oznaczająca liczbę mrówek.

W drugim wierszu standardowego wejścia znajduje się słowo długości n składające się jedynie ze znaków 'L' oraz 'P'. Jeśli i -ta litera tego słowa to 'L', to i -ta mrówka początkowo patrzy w lewo. W przeciwnym razie, gdy ta litera to 'P', mrówka ta patrzy w prawo.

Wyjście

W jedynym wierszu standardowego wyjścia powinno znaleźć się n liczb oddzielonych pojedynczymi odstępami. i -ta z tych liczb powinna być równa liczbie odbić i -tej mrówki.

Przykład

Dla danych wejściowych:

6
LPPLPL

poprawnym wynikiem jest:

0 1 3 3 2 1

Wyjaśnienie przykładu: Pierwsza mrówka patrzy początkowo w lewo i nigdy nie odbije się od żadnej innej. Ostatnia mrówka zderzy się z piątą mrówką w punkcie 5.5, po czym zacznie maszerować w prawo i już nigdy nie skończy. Trzecia mrówka, po odbiciu się od czwartej w punkcie 3.5, zacznie iść w lewo. Druga mrówka odbije się od niej w punkcie 3, po czym obróci się w lewo i nigdy nie przestanie maszerować.