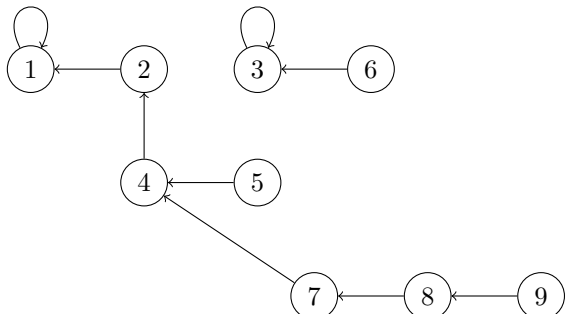


# Problem grafowy (problem-grafowy)

Limit pamięci: 32 MB

Limit czasu: 1.00 s

Jasio przygotowuje się do Bajtockiej Olimpiady Informatycznej Juniorów. Wie, że jego słabą stroną obecnie są grafy. Dlatego trenuje zadania grafowe bardzo intensywnie. Ostatnio natknął się na problem o grafach funkcyjnych: grafach, w których każdy wierzchołek ma dokładnie jedną krawędź wychodzącą. Po takim grafie łatwo nawigować, wystarczy w każdym kroku podążyć krawędzią (jedyną) w stronę następnego wierzchołka.



W zadaniu, z którym walczy teraz Jasio, trzeba dla każdego wierzchołka  $i$  wyznaczyć sumę numerów wierzchołków, na jakie natkniemy w krokach o parzystych numerach wykonując taki spacer i kończąc go w wierzchołku, którego krawędź prowadzi do samego siebie. Jasio nie wie czy to coś zmienia, ale zauważył, że we wszystkich testach w tym zadaniu dla każdego wierzchołka  $i$  jego krawędź prowadzi do wierzchołka o numerze mniejszym niż  $i$  lub równym  $i$ .

Pomóż Jasiowi rozwiązać zadanie i trenować dalej.

Napisz program, który: wczyta opis grafu, wyznaczy dla każdego wierzchołka  $i$  sumę numerów wierzchołków na odległościach parzystych od  $i$  i wypisze wyniki na standardowe wyjście.

## Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna  $N$ , określająca liczbę wierzchołków grafu. W drugim wierszu wejścia znajduje się ciąg  $N$  liczb naturalnych  $T_i$ , pooddzielanych pojedynczymi odstępami –  $i$ -ta liczba określa, że krawędź z wierzchołka numer  $i$  wychodzi do wierzchołka numer  $T_i$ .

## Wyjście

Twój program powinien wypisać na wyjście dokładnie  $N$  wierszy. W  $i$ -tym wierszu powinna się znaleźć odpowiedź dla wierzchołka  $i$ .

## Ograniczenia

$1 \leq N \leq 500\,000$ ,  $1 \leq T_i \leq i$ .

## Przykład

### Wejście

```
9
1 1 3 2 4 3 4 7 8
```

### Wyjście

```
1
2
3
5
7
6
9
13
18
```

### Wyjaśnienie

Ostatnią liczbą na wyjściu ma być  $9 + 7 + 2 = 18$ , ponieważ wierzchołki ze zbioru  $\{9, 7, 2\}$  są w parzystych odległościach od wierzchołka 9.