

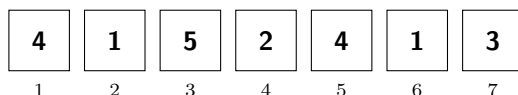
Usuwanka

XIV OIJ, zawody II stopnia
10 czerwca 2020

Kod zadania: **usu**
Limit czasu: **7 s**
Limit pamięci: **256 MB**

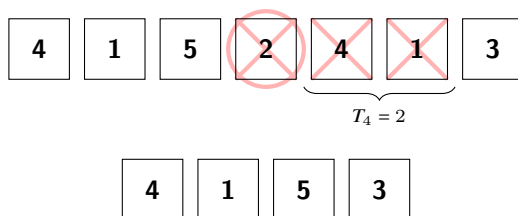


Bajtyna dostała ostatnio ciekawą grę o nazwie *Usuwanka*. Jest to gra jednoosobowa, w której gracz posługuje się zestawem N klocek ułożonych w rząd od lewej do prawej. Na każdym klocku znajduje się pewna liczba całkowita nieujemna, na i -tym klocku z lewej strony znajduje się liczba T_i . Rozważmy przykładowy zestaw $N = 7$ klocek, na których kolejno znajdują się liczby $(4, 1, 5, 2, 4, 1, 3)$.

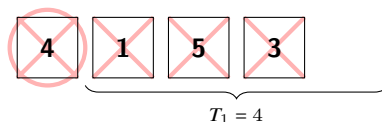


W każdej turze Bajtyna wybiera pewien klocek, usuwa go z zestawu, a razem z nim usuwa tyle sąsiadów po prawej stronie, ile wynosiła liczba na wybranym klocku. Czyli jeśli wybierze klocek na pozycji i , to usunie, oprócz niego, T_i klocek z prawej. Jeżeli klocek po prawej stronie wybranego elementu jest mniej niż T_i to wszystkie klocki po prawej są usuwane. Jeżeli jednak po prawej stronie po usunięciu klocek pozostaną jeszcze jakieś klocki, to przysuwane są one w lewo tak, aby w ciągu klocek nie powstała dziura.

Dla przykładu, jeżeli Bajtyna wybrałaby klocek na czwartej pozycji z liczbą $T_4 = 2$ to usunięty zostanie ten klocek oraz 2 klocki z jego prawej strony, to jest klocek na piątej i szóstej pozycji. Wtedy Bajtynie pozostaną klocki: $(4, 1, 5, 3)$ (jak na rysunku poniżej).



Żałujemy, że teraz Bajtyna wybrała klocek na pierwszej pozycji z liczbą $T_1 = 4$. Po prawej stronie są już jedynie 3 klocki, dlatego Bajtyna może w ten sposób usunąć wszystkie klocki.



Celem gry jest usunięcie wszystkich klocek. Bajtyna zawsze była zwolenniczką zasady, żeby nie robić więcej niż potrzeba – w tym przypadku, chciałaby osiągnąć cel gry wykonując jak najmniej ruchów. Czy pomożesz jej w tym zadaniu?

Napisz program, który wczyta początkowy ciąg klocek w grze *Usuwanka*, wyznaczy minimalną liczbę ruchów niezbędną do usunięcia wszystkich klocek i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita N ($1 \leq N \leq 200\,000$), określająca liczbę klocek. W drugim (ostatnim) wierszu wejścia znajduje się ciąg N nieujemnych liczb całkowitych T_i ($0 \leq T_i < N$), pooddzielanych pojedynczymi odstępami – są to liczby znajdujące się na kolejnych klockach od lewej do prawej.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba całkowita – minimalna liczba ruchów potrzebnych do usunięcia wszystkich klocek.



Ocenianie

Możesz rozwiązać zadanie w kilku prostszych wariantach – niektóre grupy testów spełniają pewne dodatkowe ograniczenia. Poniższa tabela pokazuje, ile punktów otrzyma Twój program, jeśli przejdzie testy z takim ograniczeniem.

Dodatkowe ograniczenia	Liczba punktów
na klockach znajdują się jedynie liczby 0 lub 1	29
$N \leq 10$	44
$N \leq 100$	55
$N \leq 1000$	67

Przykłady

Wejście dla testu usu0a:

```
7
4 1 5 2 4 1 3
```

Wyjście dla testu usu0a:

```
2
```

Wyjaśnienie do przykładu: Bajtyna może zacząć od wybrania klocka na czwartej pozycji z liczbą 2, a następnie mogłaby wybrać klocek na pierwszej pozycji liczbą 4 (zgodnie z przykładem opisanym w treści powyżej). Zauważ, że nie jest to jedyna możliwa optymalna sekwencja ruchów. Można także zacząć od wybrania klocka 5 na trzeciej pozycji, a następnie klocka 4 na pierwszej pozycji. Jeszcze inną możliwością jest wybranie klocka 4 na piątej pozycji, a następnie klocka 4 na pierwszej pozycji.

Wejście dla testu usu0b:

```
5
1 1 1 1 1
```

Wyjście dla testu usu0b:

```
3
```

Wyjaśnienie do przykładu: W tym przypadku, Bajtyna mogłaby wielokrotnie wybierać zawsze pierwszy klocek, co usuwałoby go oraz sąsiadujący klocek z prawej (o ile istnieje). A zatem po pierwszym ruchu ciąg klocków to (1, 1, 1), po drugim: (1), a po trzecim nie mamy już żadnego klocka.

Wejście dla testu usu0c:

```
5
4 4 4 4 4
```

Wyjście dla testu usu0c:

```
1
```

Wyjaśnienie do przykładu: W tym przypadku wystarczy, żeby Bajtyna wybrała pierwszy klocek. Spowoduje to usunięcie wszystkich klocków.

Wejście dla testu usu0d:

```
6
0 0 0 0 0 0
```

Wyjście dla testu usu0d:

```
6
```

Wyjaśnienie do przykładu: W tym przypadku wybory Bajtyny nie mają znaczenia, i tak musi wybrać każdy klocek tego ciągu uzyskując po drodze (0, 0, 0, 0, 0), (0, 0, 0, 0), (0, 0, 0), (0, 0), (0) oraz na końcu ciąg pusty.

Pozostałe testy przykładowe

- test usu0e: $N = 990$, ciąg liczb napisanych na klockach to (1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, ..., 44, 44, 44, 44)
- test usu0f: $N = 200\,000$, ciąg liczb napisanych na klockach to (1, 0, 1, 0, 1, 0, ...)
- test usu0g: $N = 200\,000$, ciąg liczb napisanych na klockach to (4, 1, 4, 1, 4, 1, ...)