



## Mətn redaktoru

Ramin CEOI 2024'də iştirak edir. O, bu günün ən çətin sualına olan həllini demək olar ki bitirib, həmçinin bu həllin 100 bal alacağına əmindir! Xırda bir problem qalır: bir yerdə sözü səhv yazdığını gördü. Vəziyyəti daha da pis edən budur ki, 2008'dən bəri istifadə etdiyi ən sevdiyi kompyuter siçanı nəhayət xarab oldu və işləmir. Buna görə də o düzəliş etməli olduğu yerə klaviatura vasitəsi ilə getməlidir.

Raminin proqramında uzunluqları  $l_1, l_2, \dots, l_N$  olan  $N$  sətir var. Ramin proqramını həmişə boş sətir ilə bitirir, ona görə də  $l_N = 0$ . Cursor istənilən iki simvol arasına, sətirin əvvəlinə, və sonuna yerləşdirilə bilər. Yəni  $i$ 'ci sətirdə kursoru qoya biləcəyiniz 1'dən  $l_i + 1$ 'ə nömrələnmiş  $l_i + 1$  müxtəlif pozisiya (sütun adlandırılır) var. Məsələn, 2'ci sətirin 6'cı sütununa yerləşdirilmiş cursor belə görünür:

```
1 | i | n | t | e | r | n | e | t | a | | |
2 | . | . | r | e | t | u | r | n | a | * | a |
3 | } |
4 | |
(2, 6)
```

Ramin kursoru  $s_l$ 'ci sətirin  $s_c$ 'ci sütunundan  $e_l$ 'ci sətirin  $e_c$ 'ci sütununa aparmaq istəyir. O ən az neçə dəfə düymələrə sıxmalı olduğunu bilmək istəyir.

Üfüqi ox işarələri kifayət qədər sadədir. *sol* düyməni sıxanda cursor əgər sətirin əvvəlindədirsə, əvvəlki sətirin sonuncu sütununa, əks halda soldakı sütuna gedir. Oxşar formada *sağ* düyməni sıxanda cursor əgər sətirin sonundadırsa, növbəti sətirin ilk sütununa, əks halda sağdakı sütuna gedir.

Məsələn, *sol* düyməni sıxmaq belə olur:

```
1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(3, 2) [←]

1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(3, 1) [←]

1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(2, 9) [←]
```

*sağ* düyməni sıxmaq isə belə olur:

```
1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(1, 12) [→]



1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(1, 13) [→]

1 | i | f | ( | s | e | e | n | [ | n | ] | ) |
2 | . | . | b | r | e | a | k | ; |
3 | s | e | e | n | [ | n | ] | . | = | . | t | r | u | e | ; |
4 | |
(2, 1) [→]
```



Faylın lap əvvəlində *sol* düyməni sıxmağın, və ya ən sonunda *sağ* düyməni sıxmağın heç bir təsiri olmur.

Şaquli ox işarələri bir az daha qarışıqdır. *yuxarı* düyməni sıxdıqda yuxarı sətirdə, *aşağı* düyməni sıxdıqda aşağıdakı sətirdə eyni sütuna gedir. Lakin, əgər bu əməliyyat zamanı cursor sətrin sonundan daha sağdakı bir yerə düşməli olarsa, o zaman həmin sətrin axırına gedir.

Məsələn, *yuxarı* düyməni sıxmaq belə olur:

<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>		<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>		<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>	
(3, 13)		(2, 9)		(1, 9)	

*aşağı* düyməni sıxmaq isə belə olur:

<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>		<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>		<pre> 1   i   f   .   (   s   e   e   n   [   n   ]   )   2   .   .   b   r   e   a   k   ;   3   s   e   e   n   [   n   ]   .   =   .   t   r   u   e   ;   4   </pre>	
(2, 8)		(3, 8)		(4, 1)	

Əgər *yuxarı* və ya *aşağı* düyməni sıxmaq cursoru mövcud olmayan bir sətirə aparmağa çalışarsa, cursor yerini dəyişmir.

## Giriş verilənləri

Girişin birinci sətirində  $N$  tam ədədi — Raminin sətirlərinin sayı var. İkinci sətirdə boşluqla ayrılmış  $s_l$  və  $s_c$  tam ədədləri — cursorun ilkin pozisiyası var. Oxşar şəkildə, üçüncü sətirdə getməli olduğu pozisiya  $e_l$  və  $e_c$  verilir. Dördüncü sətirdə hər sətrin uzunluğunu göstərən  $N$  sayda tam ədəd  $l_1, l_2, \dots, l_N$  verilir.

## Çıxış verilənləri

Sizin proqramınız bir sətirdə bir tam ədəd çıxışa verməlidir — cursoru  $(s_l, s_c)$ -dən  $(e_l, e_c)$ -yə aparmaq üçün lazım olan minimum əməliyyat sayı.

## Nümunələr

### Nümunə 1




Giriş verilənləri:

```
5
3 1
2 8
7 10 9 9 0
```

Çıxış verilənləri:

```
3
```

Ramin kursoru hədəf pozisiyaya *yuxarı*, *sol*, və *aşağı* düymələrini sıxmaqla üç hərəkətə apara bilər:

<pre>1 i· - =· 1 ;  2 i f· (i· =· 0 )  3 · re t u r n ;  4 a [i] · =· v ;  5  </pre>		<pre>1 i· - =· 1 ;  2 i f· (i· =· 0 )  3 · re t u r n ;  4 a [i] · =· v ;  5  </pre>		<pre>1 i· - =· 1 ;  2 i f· (i· =· 0 )  3 · re t u r n ;  4 a [i] · =· v ;  5  </pre>		<pre>1 i· - =· 1 ;  2 i f· (i· =· 0 )  3 · re t u r n ;  4 a [i] · =· v ;  5  </pre>
(3, 1)		(2, 1)		(1, 8)		(2, 8)

Alternativ olaraq, *sol*, *yuxarı* və *aşağı* düymələrini sıxmaqla da eyni sürətlə kursoru apara bilərdi. Asanlıqla göstərmək olar ki, iki əməliyyata kursoru lazımi pozisiyaya aparmaq mümkün deyil.

### Nümunə 2

Input:

```
5
1 20
3 25
25 10 40 35 0
```

Output:

```
16
```

Ən qısa yol iki dəfə *aşağı*, daha sonra 14 dəfə *sağ* düyməni sıxmaqdır.

## Məhdudiyyətlər

- $1 \leq N \leq 10^6$
- $0 \leq l_i \leq 10^9$  ( $1 \leq i \leq N$  şərtini ödəyən bütün  $i$ -lər üçün)

- $l_N = 0$
- $1 \leq s_l, e_l \leq N$
- $1 \leq s_c \leq l_{s_l} + 1$
- $1 \leq e_c \leq l_{e_l} + 1.$

## Alt tapşırıqlar

1. (5 bal)  $N \leq 2$
2. (14 bal)  $N \leq 1\,000, l_i \leq 5\,000$  ( $1 \leq i \leq N$  şərtini ödəyən bütün  $i$ 'lər üçün)
3. (26 bal)  $N \leq 1\,000$
4. (11 bal)  $l_i = l_j$  ( $1 \leq i, j \leq N - 1$  şərtini ödəyən bütün  $i$  və  $j$ 'lər üçün)
5. (44 bal) *əlavə məhdudiyyət yoxdur*