



Описка в коді

Після приголомшливої перемоги Козака Вуса на ВЮДОІ 2024, його було відправлено на CEOI 2024.

Він майже завершив свій розв'язок найважчої задачі дня. Ба більше, він майже впевнений, що отримає 100 балів! Є лише одна маленька проблема: він щойно помітив, що зробив описку в коді! Що ще гірше, його улюблена комп'ютерна миша, якою він користувався з 2008 року, здається, нарешті зламалася і відмовляється працювати. Отже, йому доведеться перейти до місця описки в коді за допомогою клавіш зі стрілками на клавіатурі.

Програма Козака Вуса має N рядків довжиною l_1, l_2, \dots, l_N . Козак Вус завжди завершує свої програми порожнім рядком, тому $l_N = 0$. Курсор може бути розташований між двома символами, а також на початку або в кінці рядка. Таким чином, рядок i має $l_i + 1$ можливих позицій для курсора (називатимемо їх колонками), нумерованих від 1 до $l_i + 1$. Наприклад, ось як виглядатиме курсор, розташований у рядку 2 в колонці 6:

```

1 | i | n | t | · | s | q | u | a | r | e | ( | i | n | t | · | a | ) | · | { |
2 | · | · | · | r | e | t | u | r | n | · | a | · | * | · | a | ; |
3 | } |
4 |
(2, 6)
```

Козак Вус хоче перемістити свій курсор з рядка s_l колонки s_c до рядка e_l колонки e_c . Він хотів би знати мінімальну кількість натискань клавіш, необхідних для цього.

Поведінка курсора при натисканні *горизонтальних* клавіш із стрілками досить проста.

Натискання клавіші *вліво* перемістить курсор до попередньої колонки, якщо курсор не знаходиться на початку рядка, в такому випадку він переміститься до кінця попереднього рядка. Аналогічно, натискання клавіші *вправо* перемістить курсор до наступної колонки, або до початку наступного рядка, якщо курсор знаходиться в кінці рядка.

Наприклад, натискання *вліво* виглядає так:

```

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)
```

Наприклад, натискання *вправо* виглядає так:

| | | | | |
|--|--|--|--|--|
| <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> |
| (1, 12) | | (1, 13) | | (2, 1) |

Натискання клавіші *вліво* на початку файлу або натискання клавіші *вправо* в кінці файлу не матиме ніякого результату.

Поведінка курсора при натисканні *вертикальних* клавіш трохи складніша.

Натискання клавіші *вгору* перемістить курсор на попередній рядок, а натискання клавіші *вниз* перемістить його на наступний рядок, залишаючи номер колонки без змін. Однак, якщо це приведе до того, що курсор *вийде за межі* нового рядка, курсор *перескочить* на кінець цього рядка.

Наприклад, натискання *вгору* виглядає так:

| | | | | |
|--|--|--|--|--|
| <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) |

Наприклад, натискання *вниз* виглядає так:

| | | | | |
|--|--|--|--|--|
| <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) |

Якщо натиснення клавіші *вгору* або *вниз* призведе до того, що курсор опиниться на рядку, якого не існує, курсор не буде переміщений.

Формат вхідних даних

Перший рядок введення містить ціле число N — кількість рядків у рішенні Козака Вуса.

Другий рядок містить два цілих числа s_l і s_c , розділені пробілом — початкова позиція курсора.

Третій рядок містить два цілих числа e_l і e_c , розділені пробілом — кінцева позиція курсора.

Четвертий рядок містить N цілих чисел, розділених пробілами, l_1, l_2, \dots, l_N — довжини кожного з рядків.

Формат вихідних даних

Ваша програма повинна вивести один рядок, що містить одне ціле число — мінімальну кількість натискань клавіш, необхідних для переміщення курсора з позиції (s_l, s_c) до (e_l, e_c) .

Приклади

Приклад 1




Формат вхідних даних:

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

Формат вихідних даних:

```
3
```

Козак Вус може перемістити курсор на кінцеву позицію за допомогою трьох натискань клавіш: спочатку *вгору*, потім *вліво* і, нарешті, *вниз*.

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

Також можна перемістити курсор на кінцеву позицію за допомогою трьох натискань клавіш: спочатку *вліво*, потім *вгору* і, нарешті, *вниз*.

Легко показати, що неможливо досягти кінцевої позиції за допомогою двох і меншої кількості натискань клавіш.

Приклад 2

Формат вхідних даних:

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

Формат вихідних даних:

```
16
```

Найкоротша можлива послідовність натискань клавіш складається з двох натискань клавіші *вниз* і чотирнадцяти натискань клавіші *вправо*.

Обмеження

- $1 \leq N \leq 10^6$
- $0 \leq l_i \leq 10^9$ (для кожного i такого, що $1 \leq i \leq N$)
- $l_N = 0$
- $1 \leq s_l, e_l \leq N$
- $1 \leq s_c \leq l_{s_l} + 1$
- $1 \leq e_c \leq l_{e_l} + 1$.

Підзадачі

1. (5 балів) $N \leq 2$
2. (14 балів) $N \leq 1\,000$, $l_i \leq 5\,000$ (для кожного i такого, що $1 \leq i \leq N$)
3. (26 балів) $N \leq 1\,000$
4. (11 балів) $l_i = l_j$ (для кожного i, j такого, що $1 \leq i, j \leq N - 1$)
5. (44 бали) без додаткових обмежень