A - Misdelivery

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 100 points

Problem Statement

Mansion AtCoder has N rooms numbered from room 1 to room N.
Each room i is inhabited by one person named S_i.

You are to deliver a package addressed to Mr./Ms. Y in room X. Determine whether the destination is correct.

Constraints

- $1 \leq N \leq 100$
- $1 \leq X \leq N$
- N and X are integers.
- S_i and Y are strings consisting of lowercase English letters with length between 1 and 10, inclusive.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
S_1 \\
S_2 \\
\vdots \\
S_N \\
X \quad Y
\end{array}
$$

Output

Print `Yes` if the name of the person living in room X is Y, and `No` otherwise.

Sample Input 1

```
3
sato
suzuki
takahashi
3 takahashi
```

Sample Output 1

```
Yes
```

The person living in room 3 is `takahashi`, which matches the name on the package.

Sample Input 2

```
3
sato
suzuki
takahashi
1 aoki
```

Sample Output 2

```
No
```

The person living in room 1 is `sato`, which does not match the name on the package, `aoki`.

Sample Input 3

```
2
smith
smith
1 smith
```

Sample Output 3

```
Yes
```

Mansion AtCoder may have people with the same name living in different rooms.

Sample Input 4

```
2
wang
li
2 wang
```

Sample Output 4

```
No
```

B - Fibonacci Reversed

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 200 points

Problem Statement

For a positive integer x, define $f(x)$ as follows:

- Let s_x be the string obtained by representing x in decimal notation (without leading zeros), and let $\mathrm{rev}(s_x)$ be the string obtained by reversing s_x. The value of $f(x)$ is the integer obtained by interpreting $\mathrm{rev}(s_x)$ as a decimal representation of an integer.

For example, when $x = 13$, we have $\mathrm{rev}(s_x) =$ `31`, so $f(x) = 31$; when $x = 10$, we have $\mathrm{rev}(s_x) =$ `01`, so $f(x) = 1$. Particularly, for any positive integer x, the value of $f(x)$ is a positive integer.

You are given positive integers X and Y. Define a sequence of positive integers $A = (a_1, a_2, \ldots, a_{10})$ as follows:

- $a_1 = X$
- $a_2 = Y$
- $a_i = f(a_{i-1} + a_{i-2})\ (i \geq 3)$

Find the value of a_{10}.

Constraints

- $1 \leq X, Y \leq 10^5$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

```
X Y
```

Output

Print the value of a_{10}.

Sample Input 1

```
1 1
```

Sample Output 1

```
415
```

The values of the elements of A are as follows:

- $a_1 = 1$
- $a_2 = 1$
- $a_3 = 2$
- $a_4 = 3$
- $a_5 = 5$
- $a_6 = 8$
- $a_7 = 31$
- $a_8 = 93$
- $a_9 = 421$
- $a_{10} = 415$

Thus, print 415.

Sample Input 2

```
3 7
```

Sample Output 2

```
895
```

Sample Input 3

```
90701 90204
```

Sample Output 3

```
9560800101
```

C - Alternated

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 350 points

Problem Statement

You are given a string S of length $2N$. S contains exactly N occurrences of A and N occurrences of B.

Find the minimum number of operations (possibly zero) needed to make S have no adjacent identical characters, where an operation consists of swapping two adjacent characters in S.

Constraints

- $1 \leq N \leq 5 \times 10^5$
- N is an integer.
- S is a string of length $2N$ consisting of N occurrences of A and N occurrences of B.

Input

The input is given from Standard Input in the following format:

N
S

Output

Print the answer.

Sample Input 1

```
3
AABBBA
```

Sample Output 1

```
2
```

By performing operations as follows, you can achieve a state with no adjacent identical characters in two operations:

- Swap the 2nd and 3rd characters. S becomes ABABBA.
- Swap the 5th and 6th characters. S becomes ABABAB.

Sample Input 2

```
3
AAABBB
```

Sample Output 2

```
3
```

Note that you can only swap adjacent characters.

Sample Input 3

```
17
AAABABABBBABABBABABABABBAAABABABBA
```

Sample Output 3

```
15
```

D - RLE Moving

Time Limit: 2 sec / Memory Limit: 1024 MiB

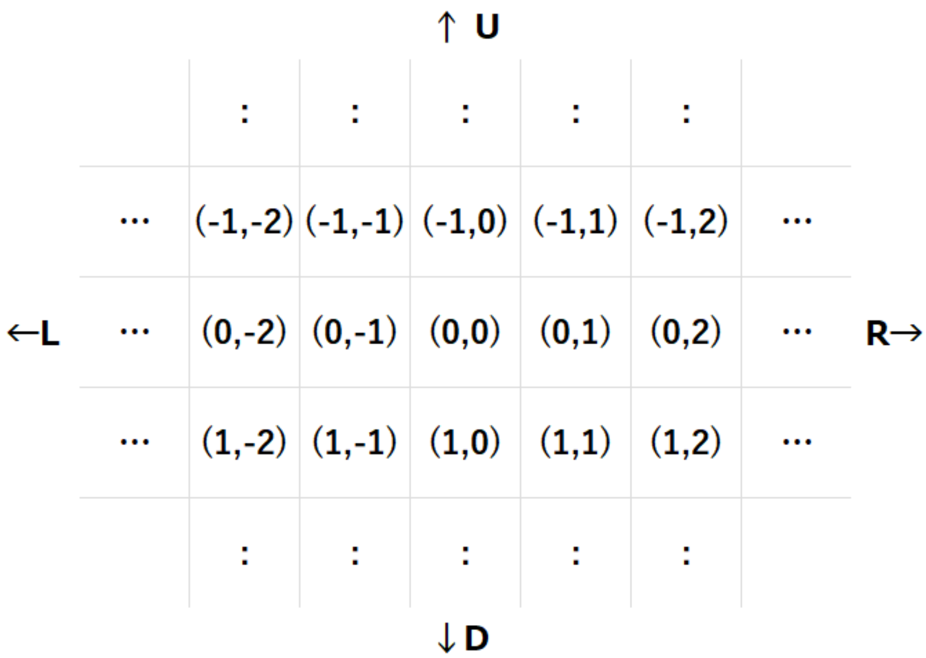
Score : 425 points

Problem Statement

There is an infinitely large grid. One cell of the grid is named cell $(0, 0)$.

The cell located r cells down and c cells right from cell $(0, 0)$ is called cell (r, c).
Here, "r cells down" means "$|r|$ cells up" when r is negative, and "c cells right" means "$|c|$ cells left" when c is negative.

Specifically, the cells around cell $(0, 0)$ are as follows:



Initially, Takahashi is at cell (R_t, C_t) and Aoki is at cell (R_a, C_a). They will each make N moves according to strings S and T of length N consisting of U, D, L, R.
For each i, Takahashi's and Aoki's i-th moves occur simultaneously: Takahashi moves one cell up if the i-th character of S is U, down if D, left if L, and right if R; Aoki moves similarly according to the i-th character of T.

Find the number of times Takahashi and Aoki are at the same cell immediately after a move during the N moves.

Since N is very large, S and T are given in the form $((S'_1, A_1), \ldots, (S'_M, A_M))$ and $((T'_1, B_1), \ldots, (T'_L, B_L))$, where S is the string obtained by concatenating "A_1 copies of character $S'_1, \ldots, A_M$ copies of character S'_M" in this order, and T is given similarly.

Constraints

- $-10^9 \leq R_t, C_t, R_a, C_a \leq 10^9$
- $1 \leq N \leq 10^{14}$
- $1 \leq M, L \leq 10^5$
- Each of S'_i and T'_i is one of U, D, L, R.
- $1 \leq A_i, B_i \leq 10^9$
- $A_1 + \cdots + A_M = B_1 + \cdots + B_L = N$
- All given values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
R_t \quad C_t \quad R_a \quad C_a \\
N \quad M \quad L \\
S'_1 \quad A_1 \\
\vdots \\
S'_M \quad A_M \\
T'_1 \quad B_1 \\
\vdots \\
T'_L \quad B_L
\end{array}
$$

Output

Print the answer.

Sample Input 1

```
0 0 4 2
3 2 1
R 2
D 1
U 3
```

Sample Output 1

```
1
```

In this case, $S =$ `RRD` and $T =$ `UUU`, and the movements proceed as follows:

- Initially, Takahashi is at cell $(0, 0)$ and Aoki is at cell $(4, 2)$.
- After the 1st move, Takahashi is at cell $(0, 1)$ and Aoki is at cell $(3, 2)$.
- After the 2nd move, Takahashi is at cell $(0, 2)$ and Aoki is at cell $(2, 2)$.
- After the 3rd move, Takahashi is at cell $(1, 2)$ and Aoki is at cell $(1, 2)$.

Thus, the number of times Takahashi and Aoki are at the same cell immediately after a move is 1.

Sample Input 2

```
1000000000 1000000000 -1000000000 -1000000000
3000000000 3 3
L 1000000000
U 1000000000
U 1000000000
D 1000000000
R 1000000000
U 1000000000
```

Sample Output 2

```
1000000001
```

From the 2000000000-th move to the 3000000000-th move, Takahashi and Aoki are at the same cell immediately after a move for 1000000001 times.

Sample Input 3

```
3 3 3 2
1 1 1
L 1
R 1
```

Sample Output 3

```
0
```

Sample Input 4

```
0 0 0 0
1 1 1
L 1
R 1
```

Sample Output 4

```
0
```

E - Yacht

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 475 points

Problem Statement

There are five six-sided dice. Each die has the numbers $A_1, \dots, A_6$ written on its faces, and each face appears with probability $\frac{1}{6}$.

You will play a single-player game using these dice with the following procedure:

1. Roll all five dice, observe the results, and **keep** any number (possibly zero) of dice.
2. Re-roll all dice that are not kept, observe the results, and additionally keep any number (possibly zero) of the re-rolled dice. **The dice kept in the previous step remain kept**.
3. Re-roll all dice that are not kept and observe the results.
4. Choose any number X. Let n be the number of dice among the five dice that show X. The score of this game is nX points.

Find the expected value of the game score when you act to maximize the expected value of the game score.

Constraints

- A_i is an integer between 1 and 100, inclusive.

Input

The input is given from Standard Input in the following format:

$$A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \quad A_6$$

Output

Print the answer. Your answer will be considered correct if the relative or absolute error from the true value is at most 10^{-5}.

Sample Input 1

```
1 2 3 4 5 6
```

Sample Output 1

```
14.6588633742
```

For example, the game may proceed as follows (not necessarily optimal):

1. Roll all five dice and get $3, 3, 1, 5, 6$. Keep the two dice that show 3.
2. Re-roll the three dice that are not kept and get $6, 6, 2$. Additionally keep the two dice that show 6.
3. Re-roll the one die that is not kept and get 4.
4. Choose $X = 6$. The dice show $3, 3, 6, 6, 4$, so the number of dice showing 6 is 2, and the score of this game is 12.

In this case, the expected value when acting optimally is $\frac{143591196865}{9795520512} = 14.6588633742\dots$.

Sample Input 2

```
1 1 1 1 1 1
```

Sample Output 2

```
5.0000000000
```

The dice may have faces with the same value written on them.

Sample Input 3

```
31 41 59 26 53 58
```

Sample Output 3

```
159.8253021021
```

F - Erase between X and Y

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 525 points

Problem Statement

There is a sequence A. Initially, $A = (0)$. (That is, A is a sequence of length 1 containing 0 as its only element).

You are given Q queries to process in order. The i-th query $(1 \leq i \leq Q)$ has one of the following forms:

- `1 x`: Insert i immediately after the location where x appears in A. Specifically, let A_j be the j-th element of the current A and n be the length of A. For p such that $A_p = x$, update A to $(A_1, \ldots, A_p, i, A_{p+1}, \ldots, A_n)$. It is guaranteed that A contains x immediately before processing this query.
- `2 x y`: Remove all elements between x and y in A, and output the sum of the values of the removed elements. Specifically, let A_j be the j-th element of the current A and n be the length of A. For p and q such that $A_p = x$ and $A_q = y$, output $A_{\min(p,q)+1} + \cdots + A_{\max(p,q)-1}$ and update A to $(A_1, \ldots, A_{\min(p,q)}, A_{\max(p,q)}, \ldots, A_n)$. It is guaranteed that A contains both x and y immediately before processing this query.

Note that for any sequence of queries, the same value never appears multiple times in A during the process of handling queries, and thus the position where a value appears in A is unique (if it exists).

Constraints

- $1 \leq Q \leq 5 \times 10^5$
- For the i-th query:
  - If it is a type 1 query:
    - $0 \leq x < i$
    - A contains x immediately before processing the query.
  - If it is a type 2 query:
    - $0 \leq x < y < i$
    - A contains both x and y immediately before processing the query.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
Q \\
\mathrm{query}_1 \\
\mathrm{query}_2 \\
\vdots \\
\mathrm{query}_Q
\end{array}
$$

Here, query_i represents the i-th query and is given in one of the following forms:

```
1 x
```

```
2 x y
```

Output

Let q be the number of type 2 queries. Output q lines. The i-th line should contain the value to be output for the i-th type 2 query.

Sample Input 1

```
6
1 0
1 1
1 0
2 2 3
1 2
2 0 5
```

Sample Output 1

```
1
5
```

Initially, $A = (0)$.

- 1st query: Insert 1 immediately after 0. A becomes $(0, 1)$.
- 2nd query: Insert 2 immediately after 1. A becomes $(0, 1, 2)$.
- 3rd query: Insert 3 immediately after 0. A becomes $(0, 3, 1, 2)$.
- 4th query: Remove the elements between 2 and 3, namely 1, and output the sum of the removed values, which is 1. A becomes $(0, 3, 2)$.
- 5th query: Insert 5 immediately after 2. A becomes $(0, 3, 2, 5)$.
- 6th query: Remove the elements between 0 and 5, namely $3, 2$, and output the sum of the removed values, which is 5. A becomes $(0, 5)$.

Sample Input 2

```
2
1 0
2 0 1
```

Sample Output 2

```
0
```

In the 2nd query, we remove all elements between 0 and 1, but there are actually no such elements, so no elements are removed and the output value is 0.

Sample Input 3

```
10
1 0
1 1
2 0 2
2 0 2
1 0
1 5
2 0 5
2 2 6
1 6
1 9
```

Sample Output 3

```
1
0
0
0
```

G - Increase to make it Increasing

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 600 points

Problem Statement

You are given a length-N integer sequence $A = (A_1, A_2, \dots, A_N)$. You are also given M pairs of integers $(L_1, R_1), (L_2, R_2), \dots, (L_M, R_M)$ $(1 \le L_i \le R_i \le N)$.

You can perform the following operation on sequence A any number of times (possibly zero):

- Choose an integer i with $1 \le i \le M$, and add 1 to each of $A_{L_i}, A_{L_i+1}, \dots, A_{R_i}$.

Determine whether it is possible to make A non-decreasing, and if possible, find the minimum number of operations required.

Constraints

- $1 \le N \le 300$
- $1 \le M \le 300$
- $1 \le A_i \le 300$
- $1 \le L_i \le R_i \le N$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad M \\
A_1 \quad A_2 \quad \dots \quad A_N \\
L_1 \quad R_1 \\
L_2 \quad R_2 \\
\vdots \\
L_M \quad R_M
\end{array}
$$

Output

If it is possible to make A non-decreasing, print the minimum number of operations required. If it is impossible, print `-1`.

Sample Input 1

```
4 3
4 2 3 2
2 2
2 3
4 4
```

Sample Output 1

```
4
```

For example, by performing operations four times as follows, you can make A non-decreasing:

- Choose $i = 1$ and perform the operation. A becomes $(4, 3, 3, 2)$.
- Choose $i = 3$ and perform the operation. A becomes $(4, 3, 3, 3)$.
- Choose $i = 3$ and perform the operation. A becomes $(4, 3, 3, 4)$.
- Choose $i = 2$ and perform the operation. A becomes $(4, 4, 4, 4)$.

Conversely, it is impossible to make A non-decreasing with three or fewer operations. Thus, print 4.

Sample Input 2

```
3 2
3 1 2
2 2
1 2
```

Sample Output 2

```
-1
```

No matter how you perform operations, it is impossible to make A non-decreasing.

Sample Input 3

```
4 4
1 1 2 3
1 1
2 2
3 3
4 4
```

Sample Output 3

```
0
```

A is already non-decreasing, so no operations are needed.

Sample Input 4

```
8 12
35 29 36 88 58 15 25 99
5 5
1 6
3 8
8 8
4 8
7 7
5 7
3 3
2 6
1 6
6 7
5 7
```

Sample Output 4

```
79
```