A - Print 341

Time Limit: 2 sec / Memory Limit: 1024 MB

Score: 100 points

Problem Statement

Given a positive integer N, print a string of N zeros and $N+1$ ones where `0` and `1` alternate.

Constraints

- N is an integer.
- $1 \leq N \leq 100$

Input

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

```
N
```

Output

Print the answer.

Sample Input 1

```
4
```

Sample Output 1

```
101010101
```

A string of four zeros and five ones where `0` and `1` alternate is `101010101`.

Sample Input 2

```
1
```

Sample Output 2

```
101
```

Sample Input 3

```
10
```

Sample Output 3

```
101010101010101010101
```

B - Foreign Exchange

Time Limit: 2 sec / Memory Limit: 1024 MB

Score: 150 points

Problem Statement

There are N countries numbered 1 to N. For each $i = 1, 2, \ldots, N$, Takahashi has A_i units of the currency of country i.

Takahashi can repeat the following operation any number of times, possibly zero:

- First, choose an integer i between 1 and $N-1$, inclusive.
- Then, if Takahashi has at least S_i units of the currency of country i, he performs the following action once:
  - Pay S_i units of the currency of country i and gain T_i units of the currency of country $(i+1)$.

Print the maximum possible number of units of the currency of country N that Takahashi could have in the end.

Constraints

- All input values are integers.
- $2 \leq N \leq 2 \times 10^5$
- $0 \leq A_i \leq 10^9$
- $1 \leq T_i \leq S_i \leq 10^9$

Input

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

$$
\begin{array}{l}
N \\
A_1 \quad A_2 \quad \ldots \quad A_N \\
S_1 \quad T_1 \\
S_2 \quad T_2 \\
\vdots \\
S_{N-1} \quad T_{N-1}
\end{array}
$$

Output

Print the answer.

Sample Input 1

```
4
5 7 0 3
2 2
4 3
5 2
```

Sample Output 1

```
5
```

In the following explanation, let the sequence $A = (A_1, A_2, A_3, A_4)$ represent the numbers of units of the currencies of the countries Takahashi has. Initially, $A = (5, 7, 0, 3)$.

Consider performing the operation four times as follows:

- Choose $i = 2$, pay four units of the currency of country 2, and gain three units of the currency of country 3. Now, $A = (5, 3, 3, 3)$.
- Choose $i = 1$, pay two units of the currency of country 1, and gain two units of the currency of country 2. Now, $A = (3, 5, 3, 3)$.
- Choose $i = 2$, pay four units of the currency of country 2, and gain three units of the currency of country 3. Now, $A = (3, 1, 6, 3)$.
- Choose $i = 3$, pay five units of the currency of country 3, and gain two units of the currency of country 4. Now, $A = (3, 1, 1, 5)$.

At this point, Takahashi has five units of the currency of country 4, which is the maximum possible number.

Sample Input 2

```
10
32 6 46 9 37 8 33 14 31 5
5 5
3 1
4 3
2 2
3 2
3 2
4 4
3 3
3 1
```

Sample Output 2

```
45
```

C - Takahashi Gets Lost

Time Limit: 3 sec / Memory Limit: 1024 MB

Score: 250 points

Problem Statement

There is a grid with H rows and W columns.

Each cell of the grid is **land** or **sea**, which is represented by H strings $S_1, S_2, \ldots, S_H$ of length W. Let (i, j) denote the cell at the i-th row from the top and j-th column from the left, and (i, j) is land if the j-th character of S_i is `.`, and (i, j) is sea if the character is `#`.

The constraints guarantee that all cells on the perimeter of the grid (that is, the cells (i, j) that satisfy at least one of $i = 1, i = H, j = 1, j = W$) are sea.

Takahashi's spaceship has crash-landed on a cell in the grid. Afterward, he moved N times on the grid following the instructions represented by a string T of length N consisting of `L`, `R`, `U`, and `D`. For $i = 1, 2, \ldots, N$, the i-th character of T describes the i-th move as follows:

- `L` indicates a move of one cell to the left. That is, if he is at (i, j) before the move, he will be at $(i, j-1)$ after the move.
- `R` indicates a move of one cell to the right. That is, if he is at (i, j) before the move, he will be at $(i, j+1)$ after the move.
- `U` indicates a move of one cell up. That is, if he is at (i, j) before the move, he will be at $(i-1, j)$ after the move.
- `D` indicates a move of one cell down. That is, if he is at (i, j) before the move, he will be at $(i+1, j)$ after the move.

It is known that all cells along his path (including the cell where he crash-landed and the cell he is currently on) are not sea. Print the number of cells that could be his current position.

Constraints

- H, W, and N are integers.
- $3 \leq H, W \leq 500$
- $1 \leq N \leq 500$
- T is a string of length N consisting of `L`, `R`, `U`, and `D`.
- S_i is a string of length W consisting of `.` and `#`.
- There is at least one cell that could be Takahashi's current position.
- All cells on the perimeter of the grid are sea.

Input

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

$$
\begin{array}{l}
H \quad W \quad N \\
T \\
S_1 \\
S_2 \\
\vdots \\
S_H
\end{array}
$$

Output

Print the answer.

Sample Input 1

```
6 7 5
LULDR
#######
#...#.#
##...##
#.#...#
#...#.#
#######
```

Sample Output 1

```
2
```

The following two cases are possible, so there are two cells that could be Takahashi's current position: $(3, 4)$ and $(4, 5)$.

- He crash-landed on cell $(3, 5)$ and moved $(3, 5) \to (3, 4) \to (2, 4) \to (2, 3) \to (3, 3) \to (3, 4)$.
- He crash-landed on cell $(4, 6)$ and moved $(4, 6) \to (4, 5) \to (3, 5) \to (3, 4) \to (4, 4) \to (4, 5)$.

Sample Input 2

```
13 16 9
ULURDLURD
################
##..##.#..####.#
###.#..#.....#.#
#..##..#####.###
#...#..#......##
###.##.#..#....#
##.#####....##.#
###.###.#.#.#..#
######.....##..#
#...#.#.######.#
##..###..#..#.##
#...#.#.#...#..#
################
```

Sample Output 2

```
6
```

D - Only one of two

Time Limit: 2 sec / Memory Limit: 1024 MB

Score: 400 points

Problem Statement

You are given three positive integers N, M, and K. Here, N and M are different.

Print the K-th smallest positive integer divisible by **exactly one** of N and M.

Constraints

- $1 \leq N, M \leq 10^8$
- $1 \leq K \leq 10^{10}$
- $N \neq M$
- N, M, and K are integers.

Input

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

```
N M K
```

Output

Print the K-th smallest positive integer divisible by exactly one of N and M.

Sample Input 1

```
2 3 5
```

Sample Output 1

```
9
```

The positive integers divisible by exactly one of 2 and 3 are $2, 3, 4, 8, 9, 10, \ldots$ in ascending order.

Note that 6 is not included because it is divisible by both 2 and 3.

The fifth smallest positive integer that satisfies the condition is 9, so we print 9.

Sample Input 2

```
1 2 3
```

Sample Output 2

```
5
```

The numbers that satisfy the condition are $1, 3, 5, 7, \ldots$ in ascending order.

Sample Input 3

```
100000000 99999999 10000000000
```

Sample Output 3

```
500000002500000000
```

E - Alternating String

Time Limit: 3 sec / Memory Limit: 1024 MB

Score: 450 points

Problem Statement

A string consisting of `0` and `1` is called a **good string** if two consecutive characters in the string are always different.

You are given a string S of length N consisting of `0` and `1`. Q queries will be given and must be processed in order.

There are two types of queries:

- `1 L R`: Flip each of the L-th to R-th characters of S. That is, for each integer i satisfying $L \leq i \leq R$, change the i-th character of S to `0` if it is `1`, and vice versa.
- `2 L R`: Let S' be the string of length $(R-L+1)$ obtained by extracting the L-th to R-th characters of S (without changing the order). Print `Yes` if S' is a good string and `No` otherwise.

Constraints

- $1 \leq N, Q \leq 5 \times 10^5$
- S is a string of length N consisting of `0` and `1`.
- $1 \leq L \leq R \leq N$ for queries of types 1 and 2.
- There is at least one query of type 2.
- $N, Q, L,$ and R are integers.

Input

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

$$
\begin{array}{l}
N \quad Q \\
S \\
query_1 \\
query_2 \\
\vdots \\
query_Q
\end{array}
$$

Each query $query_i$ $(1 \leq i \leq Q)$ is given in the form:

$$1 \quad L \quad R$$

or:

$$2 \quad L \quad R$$

Output

Let K be the number of queries of type 2. Print K lines.

The i-th line should contain the response to the i-th query of type 2.

Sample Input 1

```
5 6
10100
2 1 3
2 1 5
1 1 4
2 1 5
1 3 3
2 2 4
```

Sample Output 1

```
Yes
No
Yes
No
```

Initially, $S =$`10100`. When processing the queries in the order they are given, the following occurs:

- For the first query, the string obtained by extracting the 1-st to 3-rd characters of S is $S' =$`101`. This is a good string, so print `Yes`.
- For the second query, the string obtained by extracting the 1-st to 5-th characters of S is $S' =$ `10100`. This is not a good string, so print `No`.
- For the third query, flip each of the 1-st to 4-th characters of S. The string S becomes $S =$`01010`.
- For the fourth query, the string obtained by extracting the 1-st to 5-th character of S is $S' =$`01010`. This is a good string, so print `Yes`.
- For the fifth query, flip the 3-rd character of S. The string S becomes $S =$`01110`.
- For the sixth query, the string obtained by extracting the 2-nd to 4-th character of S is $S' =$`111`. This is not a good string, so print `No`.

Sample Input 2

```
1 2
1
1 1 1
2 1 1
```

Sample Output 2

```
Yes
```

Note that a string of a single character `0` or `1` satisfies the condition of being a good string.

F - Breakdown

Time Limit: 2 sec / Memory Limit: 1024 MB

Score: 475 points

Problem Statement

You are given a simple undirected graph consisting of N vertices and M edges. For $i = 1, 2, \ldots, M$, the i-th edge connects vertices u_i and v_i. Also, for $i = 1, 2, \ldots, N$, vertex i is assigned a positive integer W_i, and there are A_i pieces placed on it.

As long as there are pieces on the graph, repeat the following operation:

- First, choose and remove one piece from the graph, and let x be the vertex on which the piece was placed.
- Choose a (possibly empty) set S of vertices adjacent to x such that $\sum_{y \in S} W_y < W_x$, and place one piece on each vertex in S.

Print the maximum number of times the operation can be performed.

It can be proved that, regardless of how the operation is performed, there will be no pieces on the graph after a finite number of iterations.

Constraints

- All input values are integers.
- $2 \leq N \leq 5000$
- $1 \leq M \leq \min\{N(N-1)/2, 5000\}$
- $1 \leq u_i, v_i \leq N$
- $u_i \neq v_i$
- $i \neq j \implies \{u_i, v_i\} \neq \{u_j, v_j\}$
- $1 \leq W_i \leq 5000$
- $0 \leq A_i \leq 10^9$

Input

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

$$
\begin{array}{l}
N \quad M \\
u_1 \quad v_1 \\
u_2 \quad v_2 \\
\vdots \\
u_M \quad v_M \\
W_1 \ W_2 \ \ldots \ W_N \\
A_1 \ A_2 \ \ldots \ A_N
\end{array}
$$

Output

Print the answer.

Sample Input 1

```
6 6
1 2
2 3
3 1
3 4
1 5
5 6
9 2 3 1 4 4
1 0 0 0 0 1
```

Sample Output 1

```
5
```

In the following explanation, let $A = (A_1, A_2, \ldots, A_N)$ represent the numbers of pieces on the vertices. Initially, $A = (1, 0, 0, 0, 0, 1)$.

Consider performing the operation as follows:

- Remove one piece from vertex 1 and place one piece each on vertices 2 and 3. Now, $A = (0, 1, 1, 0, 0, 1)$.
- Remove one piece from vertex 2. Now, $A = (0, 0, 1, 0, 0, 1)$.
- Remove one piece from vertex 6. Now, $A = (0, 0, 1, 0, 0, 0)$.
- Remove one piece from vertex 3 and place one piece on vertex 2. Now, $A = (0, 1, 0, 0, 0, 0)$.
- Remove one piece from vertex 2. Now, $A = (0, 0, 0, 0, 0, 0)$.

In this procedure, the operation is performed five times, which is the maximum possible number of times.

Sample Input 2

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

Sample Output 2

```
0
```

In this sample input, there are no pieces on the graph from the beginning.

Sample Input 3

```
10 20
4 8
1 10
1 7
5 9
9 10
8 10
7 5
1 4
7 3
8 7
2 8
5 8
4 2
5 1
7 2
8 3
3 4
8 9
7 10
2 3
25 5 1 1 16 5 98 3 21 1
35 39 32 11 35 37 14 29 36 1
```

Sample Output 3

```
1380
```

G - Highest Ratio

Time Limit: 2 sec / Memory Limit: 1024 MB

Score: 575 points

Problem Statement

You are given a sequence $A = (A_1, A_2, \ldots, A_N)$ of length N.
For each $k = 1, 2, \ldots, N$, solve the following problem:

- Find the maximum possible average value of the k-th to r-th terms of the sequence A when choosing an integer r such that $k \leq r \leq N$.

  Here, the average value of the k-th to r-th term of the sequence A is defined as $\frac{1}{r-k+1} \sum_{i=k}^{r} A_i$.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 10^6$
- All input values are integers.

Input

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

$$\begin{array}{l} N \\ A_1 \quad A_2 \quad \ldots \quad A_N \end{array}$$

Output

Print N lines.
The i-th line $(1 \leq i \leq N)$ should contain the answer to the problem for $k = i$.
Your output will be considered correct if, for every line, the absolute or relative error of the printed value from the true value is at most 10^{-6}.

Sample Input 1

```
5
1 1 4 5 3
```

Sample Output 1

```
2.80000000
3.33333333
4.50000000
5.00000000
3.00000000
```

For $k = 1$, the possible choices for r are $r = 1, 2, 3, 4, 5$, and the average value for each of them is:

- $\frac{1}{1} = 1$
- $\frac{1}{2}(1 + 1) = 1$
- $\frac{1}{3}(1 + 1 + 4) = 2$
- $\frac{1}{4}(1 + 1 + 4 + 5) = 2.75$
- $\frac{1}{5}(1 + 1 + 4 + 5 + 3) = 2.8$

Thus, the maximum is achieved when $r = 5$, and the answer for $k = 1$ is 2.8.
Similarly, for $k = 2, 3, 4, 5$, the maximum is achieved when $r = 4, 4, 4, 5$, respectively, with the values of $\frac{10}{3} = 3.333\ldots$, $\frac{9}{2} = 4.5$, $\frac{5}{1} = 5$, $\frac{3}{1} = 3$.

Sample Input 2

```
3
999999 1 1000000
```

Sample Output 2

```
999999.00000000
500000.50000000
1000000.00000000
```