A - Odd Position Sum

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 100 points

Problem Statement

You are given a sequence of positive integers of length N: $A = (A_1, A_2, \dots, A_N)$.

Find the sum of the odd-indexed elements of A. That is, find $A_1 + A_3 + A_5 + \dots + A_m$, where m is the largest odd number not exceeding N.

Constraints

- $1 \le N \le 100$
- $1 \le A_i \le 100$
- All input values are integers.

Input

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

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

Output

Print the answer.

Sample Input 1

Copy

```
7
3 1 4 1 5 9 2
```

Sample Output 1

Copy

```
14
```

The sum of the odd-indexed elements of A is $A_1 + A_3 + A_5 + A_7 = 3 + 4 + 5 + 2 = 14$.

Sample Input 2

Copy

```
1
100
```

Sample Output 2

Copy

```
100
```

Sample Input 3

Copy

```
14
100 10 1 10 100 10 1 10 100 10 1 10 100 10
```

Sample Output 3

Copy

```
403
```

B - Four Hidden

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 250 points

Problem Statement

You are given a string T consisting of lowercase English letters and `?`, and a string U consisting of lowercase English letters.

The string T is obtained by taking some lowercase-only string S and replacing exactly four of its characters with `?`.

Determine whether it is possible that the original string S contained U as a contiguous substring.

Constraints

- T is a string of length between 4 and 10, inclusive, consisting of lowercase letters and `?`.
- T contains exactly four occurrences of `?`.
- U is a string of length between 1 and $|T|$, inclusive, consisting of lowercase letters.

Input

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

$$T$$
$$U$$

Output

Print `Yes` if it is possible that the original string S contained U as a contiguous substring; otherwise, print `No`.

Sample Input 1

Copy

```
tak??a?h?
nashi
```

Sample Output 1

Copy

```
Yes
```

For example, if S is `takanashi`, it contains `nashi` as a contiguous substring.

Sample Input 2

Copy

```
??e??e
snuke
```

Sample Output 2

Copy

```
No
```

No matter what characters replace the `?` s in T, S cannot contain `snuke` as a contiguous substring.

Sample Input 3

Copy

```
????
aoki
```

Sample Output 3

Copy

```
Yes
```

C - 403 Forbidden

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 300 points

Problem Statement

There are N users on WAtCoder, numbered from 1 to N, and M contest pages, numbered from 1 to M. Initially, no user has view permission for any contest page.

You are given Q queries to process in order. Each query is of one of the following three types:

- `1 X Y` : Grant user X view permission for contest page Y.
- `2 X` : Grant user X view permission for all contest pages.
- `3 X Y` : Answer whether user X can view contest page Y.

It is possible for a user to be granted permission for the same contest page multiple times.

Constraints

- $1 \le N \le 2 \times 10^5$
- $1 \le M \le 2 \times 10^5$
- $1 \le Q \le 2 \times 10^5$
- $1 \le X \le N$
- $1 \le Y \le M$
- All input values are integers.

Input

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

$$\begin{array}{l} N \quad M \quad Q \\ \text{query}_1 \\ \text{query}_2 \\ \vdots \\ \text{query}_Q \end{array}$$

Each query_i is in one of the following formats:

$$1 \quad X \quad Y$$

$$2 \quad X$$

$$3 \quad X \quad Y$$

Output

For each query of the third type, print `Yes` if user X can view contest page Y, otherwise print `No`, each on its own line.

Sample Input 1

Copy

```
2 3 5
1 1 2
3 1 1
3 1 2
2 2
3 2 3
```

Sample Output 1

Copy

```
No
Yes
Yes
```

- In the first query, user 1 is granted permission to view contest page 2.
- At the second query, user 1 can view only page 2; they cannot view page 1, so print `No`.
- At the third query, user 1 can view page 2, so print `Yes`.
- In the fourth query, user 2 is granted permission to view all pages.
- At the fifth query, user 2 can view pages $1, 2, 3$; they can view page 3, so print `Yes`.

Sample Input 2 Copy

```
5 5 10
2 2
3 4 4
1 1 1
1 4 1
1 4 2
1 4 4
1 2 4
3 3 2
3 5 4
3 2 1
```

Sample Output 2 Copy

```
No
No
No
Yes
```

D - Forbidden Difference

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 425 points

Problem Statement

You are given a length-N integer sequence $A = (A_1, A_2, \ldots, A_N)$ and a non-negative integer D. We wish to delete as few elements as possible from A to obtain a sequence B that satisfies the following condition:

- $|B_i - B_j| \neq D$ for all i, j $(1 \leq i < j \leq |B|)$.

Find the minimum number of deletions required.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $0 \leq D \leq 10^6$
- $0 \leq A_i \leq 10^6$
- All input values are integers.

Input

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

$$\begin{array}{llll} N & D & & \\ A_1 & A_2 & \ldots & A_N \end{array}$$

Output

Print the answer.

Sample Input 1

Copy

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

Sample Output 1

Copy

```
1
```

Deleting $A_1 = 3$ yields $B = (1, 4, 1, 5)$, which satisfies $|B_i - B_j| \neq 2$ for all $i < j$.

Sample Input 2 Copy

```
4 3
1 6 1 8
```

Sample Output 2 Copy

```
0
```

The sequence A may already satisfy the condition.

Sample Input 3 Copy

```
10 3
1 6 2 10 2 3 2 10 6 4
```

Sample Output 3 Copy

```
2
```

E - Forbidden Prefix

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 500 points

Problem Statement

There are two multisets of strings, X and Y, both initially empty.

You are given Q queries to process in order. In the i-th query, you receive an integer T_i and a string S_i. If $T_i = 1$, insert S_i into X; if $T_i = 2$, insert S_i into Y.

After processing each query, print this value:

- the number of strings in Y that have no element of X as a prefix.

Constraints

- Q is an integer between 1 and 2×10^5, inclusive.
- $T_i \in \{1, 2\}$
- Each S_i is a string of length between 1 and 5×10^5, inclusive, consisting of lowercase English letters.
- $\displaystyle\sum_{i=1}^{Q} |S_i| \leq 5 \times 10^5$

Input

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

$$\begin{array}{l} Q \\ T_1 \quad S_1 \\ T_2 \quad S_2 \\ \vdots \\ T_Q \quad S_Q \end{array}$$

Output

Print Q lines. The i-th line $(1 \leq i \leq Q)$ should contain the count after processing the i-th query.

Sample Input 1

Copy

```
4
1 at
2 watcoder
2 atcoder
1 wa
```

Sample Output 1

Copy

```
0
1
1
0
```

The counts after processing the queries for $i = 1, 2, 3, 4$ are as follows.

- $i = 1$: Y is empty, so the count is 0.
- $i = 2$: `watcoder` has no element of X as a prefix, so the count is 1.
- $i = 3$: `watcoder` has no element of X as a prefix, while `atcoder` has `at` as a prefix, so the count is 1.
- $i = 4$: `watcoder` has `wa` as a prefix, and `atcoder` has `at` as a prefix, so the count is 0.

Sample Input 2

Copy

```
10
1 w
1 avko
2 atcoder
1 bzginn
2 beginner
1 atco
2 contest
1 ntxcdg
1 atc
1 contest
```

Sample Output 2

Copy

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

F - Shortest One Formula

Time Limit: 2 sec / Memory Limit: 1024 MB

Score : 500 points

Problem Statement

You are given a positive integer N.

Among all valid arithmetic expressions consisting of the characters `1` , `+` , `*` , `(` , and `)` , find one of the minimum length whose value is N.

More formally, among the strings S satisfying all of the following conditions, find one of the minimum length:

- S conforms to the symbol `<expr>` in the BNF (https://en.wikipedia.org/wiki/Backus%E2%80%93Naur_form) below.
- The value of the expression represented by S is N.

```
<expr>   ::= <term> | <expr> "+" <term>
<term>   ::= <factor> | <term> "*" <factor>
<factor> ::= <number> | "(" <expr> ")"
<number> ::= "1" | "1" <number>
```

Strings that conform to `<expr>` include:

- `1111+111` representing $1111 + 111$.
- `(1+1)*(1+1)` representing $(1 + 1) \times (1 + 1)$.
- `(11+(1+1)*(1+1))+1` representing $(11 + (1 + 1) \times (1 + 1)) + 1$.

Strings that do not conform to `<expr>` include:

- `(1+1)(1+1)`
- `1+2`
- `1-1`
- `1/1`
- `)1(`
- `1++1`
- `+1`
- `(+1)`
- `1*+1`

Constraints

- $1 \leq N \leq 2000$
- All input values are integers.

Input

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

```
N
```

Output

Print a solution.

Sample Input 1 Copy

```
9
```

Sample Output 1 Copy

```
(1+1+1)*(1+1+1)
```

Expressions whose value is 9 include:

- `(1+1+1)*(1+1+1)`
- `1+1+1+1+1+1+1+1+1`
- `(1+1)*(1+1)*(1+1)+1`

Among them, a shortest is `(1+1+1)*(1+1+1)` .

Sample Input 2 Copy

```
11
```

Sample Output 2 Copy

```
11
```

Sample Input 3 Copy

```
403
```

Sample Output 3

Copy

```
1+(1+1+1)*(1+11+11+111)
```

G - Odd Position Sum Query

Time Limit: 4 sec / Memory Limit: 1024 MB

Score : 600 points

Problem Statement

There is an initially empty sequence A.

You are given Q queries to process in order. The i-th query is explained below:

> You are given an integer y_i. If $i = 1$, let z be 0; otherwise, let z be the answer to the $(i-1)$-th query. Define $x_i = ((y_i + z) \bmod 10^9) + 1$. Append x_i to the end of A.
> Then, let $B = (B_1, B_2, \ldots, B_i)$ be the sequence A sorted in ascending order, and find the sum of the odd-indexed elements of B. That is, find $B_1 + B_3 + B_5 + \cdots + B_m$, where m is the largest odd number not exceeding i.

Constraints

- $1 \leq Q \leq 3 \times 10^5$
- $0 \leq y_i < 10^9$
- $1 \leq x_i \leq 10^9$
- All input values are integers.

Input

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

Q
y_1
y_2
$\vdots$
y_Q

Output

Print Q lines. The i-th line should contain the answer to the i-th query.

Sample Input 1

Copy

```
5
1
3
1
999999994
999999993
```

Sample Output 1

Copy

```
2
2
8
6
1000000006
```

- For the 1st query, $y_1 = 1, z = 0$, so $x_1 = ((1+0) \bmod 10^9) + 1 = 2$. Appending this to the end of A gives $A = (2)$. Sorting A in ascending order yields $B = (2)$, and the sought value is $B_1 = 2$.
- For the 2nd query, $y_2 = 3, z = 2$, so $x_2 = ((3+2) \bmod 10^9) + 1 = 6$. Appending gives $A = (2, 6)$, so $B = (2, 6)$ and the sought value is $B_1 = 2$.
- For the 3rd query, $y_3 = 1, z = 2$, so $x_3 = ((1+2) \bmod 10^9) + 1 = 4$. Appending gives $A = (2, 6, 4)$, so $B = (2, 4, 6)$ and the sought value is $B_1 + B_3 = 8$.
- For the 4th query, $y_4 = 999999994, z = 8$, so $x_4 = ((999999994 + 8) \bmod 10^9) + 1 = 3$. Appending gives $A = (2, 6, 4, 3)$, so $B = (2, 3, 4, 6)$ and the sought value is $B_1 + B_3 = 6$.
- For the 5th query, $y_5 = 999999993, z = 6$, so $x_5 = ((999999993 + 6) \bmod 10^9) + 1 = 1000000000$. Appending gives $A = (2, 6, 4, 3, 1000000000)$, so $B = (2, 3, 4, 6, 1000000000)$ and the sought value is $B_1 + B_3 + B_5 = 1000000006$.

Sample Input 2

Copy

```
8
105282053
695234822
468007124
120710491
568831200
700753895
765188109
262666319
```

Sample Output 2

Copy

```
105282054
105282054
905798931
599798602
995656103
891549225
1652393438
1652393438
```

Below are the values of $x_1, x_2, \ldots, x_8$ in order:

```
105282054
800516877
573289179
26509423
168629803
696409999
656737335
915059758
```