

Move Union Find

In this problem you are to implement a variant of Union Find: *Move Union Find*.

The data structure you need to write is also a collection of disjoint sets, supporting 3 operations:

operation	description
1 a b	Union the sets containing a and b . If a and b are already in the same set, ignore this command
2 a b	Move the element a to the set containing b . If a and b are already in the same set, ignore this command
3 a	Return the number of elements and the sum of elements in the set containing a

Initially, the collection contains n sets: {1}, {2}, {3}, ..., {N}

Input

The first line contains two integer N and M. N is the total number of sets initially. M is the number of commands to follow. $1 \leq N, M \leq 100,000$

Each of the next M lines contain a command. For each operation, $1 \leq a, b \leq N$.

Output

For each type-3 command, output 2 integers: the number of elements and the sum of elements.

Example 1

Input :

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

Output :

```
3 7
```

Explanation:

Initial sets: {1}, {2}, {3}, {4}, {5}

After 1 1 2 : {1, 2}, {3}, {4}, {5}

After 2 3 4 : {1, 2}, {3, 4}, {5}

After 1 3 5 : {1, 2}, {3, 4, 5}

After 2 4 1 : {1, 2, 4}, {3, 5}

Example 2

Input :

5 6

1 1 2

2 3 2

3 3

2 1 4

2 2 4

3 3

Output :

3 6

1 3

Prime Pair

A prime pair is a pair of the form $(p, p + 2)$ where p and $(p + 2)$ are both prime. The first few prime pairs are (3, 5), (5, 7), (11, 13). In this problem, you are given an integer N and you should find the N 'th prime pair.

Input

The input will contain a single integer N ($1 \leq N \leq 100,000$).

Output

The output should be a single **line**, the N 'th prime pair formatted as **(p1, p2)** (there is a space after the comma). It is guaranteed that primes in the 100,000th prime pair are less than 20,000,000.

Example 1

Input:

1

Output:

(3, 5)

Example 2

Input:

2

Output:

(5, 7)

Example 3

Input:

3

Output:

(11, 13)

Example 4

Input:

4

Output:

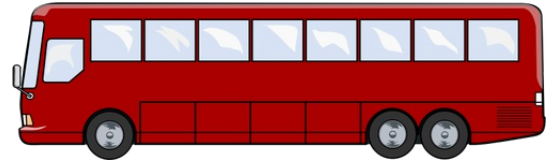
(17, 19)

Bus Tour

Hangzhou is a beautiful city for tourist. The buses with numbers that begin with 'Y' go to scenic spots, train stations, tourist centers and bus stations (all the things that the tourists need). Y1 is a bus route starting from Lingyin Temple and around the West Lake. There is N stops for Y1.

The scenery between two adjacent stops can be measured by a integer scale of "niceness". Positive niceness value indicates nice view and negative value indicates the scenery between stops is dull.

Macro Pool is a reporter for *Tourists' World* who arrived in Hangzhou last week. His task is to find two different stops along the Y1 bus route such that the niceness score of all the segments in between them is the largest.



Can you help him find those stops?

Input

The first line of input contains an integer, N, the number of stops along the Y1 bus route, $2 \leq N \leq 20,000$

Each of the next N-1 lines contains a single integer. The i-th integer indicating niceness between stop i and stop i+1.

The absolute value of niceness will not exceed 10^9 .

Output If the maximum possible sum between two stops is not positive, your program should print a line:

"Yet another overrated tourist destination"

Otherwise, your program should identify the beginning bus stop i and the ending bus stop j that identify the segment of the route which yields the maximal sum of niceness.

If more than one segment is equally maximally nice, choose the one with the longest bus ride (largest number of stops, $j - i$). To break ties further in longest maximal segments, choose the segment that begins with the earliest stop (lowest i).

Print a line in the form:

"The nicest part of Y1 is between stops i and j"

Example 1

Input

3
-2
5

Output

The nicest part of Y1 is between stops 2 and 3

Example 2

Input

4
-1
-1
-1

Output

Yet another overrated tourist destination

Scorewords

Alice developed a new word game called *Scorewords*. The game comes with a special dictionary that matches words to their point values. In the game, each player comes up with a sentence. The sentences are scored based on the sum of the scores for each word. If a word does not appear in the dictionary, its score is zero.

Alice contracted you to write a program that given all the sentences provided by different players, calculates their scores.

Input

The first line of input contains 2 positive integers, $m \leq 1000$, the number of words in the dictionary, and $n \leq 100$, the number of sentences.

m lines follow; each containing a word (a string of up to 16 lower-case letters) and its point value ($0 \leq \text{value} \leq 1,000,000$).

Following the dictionary are the n sentences. Each sentence consists of one or more lines of text such that each space separated word contains only lowercase letters. A period on a line by itself marks the end of the sentence.

Output For each sentence, output its score computed as the sum of values for all words that appear in the sentence. Words that do not appear in the dictionary have scores equal to zero. The output ends with a newline.

Example

Input:

```
7 2
administer 100000
spending 200000
manage 50000
responsibility 25000
expertise 100
skill 50
money 75000
the incumbent will administer the spending of kindergarden milk money
and exercise responsibility for making change he or she will share
responsibility for the task of managing the money with the assistant
whose skill and expertise shall ensure the successful spending exercise
.
this individual must have the skill to perform a heart transplant and
expertise in rocket science
.
```

Output:

```
700150
150
```

Shannon's database

Shannon is working on her database project where she is required to write a code to generate the original data from the corrupted data. The data is a list of non-decimal numerals. For each input, she will get a key using which she can generate the original list. The key only contains two characters **R** and **D**. Character **R** reverses the input list whereas character **D** drops the first element of the list. Shannon knows that following the above algorithm on the corrupted list, she can generate the original list. There are a few cases where the data is irrecoverable in which case she has to print **error**. Cases like these might arrive when she is applying **D** on an empty list. Help Shannon generate the original data.

Input

One line with a key $k(1 \leq \text{length}(k) \leq 10^5)$ consisting of characters 'R' and 'D'. One line with an int $n(0 \leq n \leq 10^5)$ which denotes the number of elements in the input list. One line with a list of n int. The list is of the form $[y_1, \dots, y_n]$, where $(1 \leq y_i \leq 100)$

Output

One line containing either the resulting list or "error". The resulting list should start with **[** and end with **]**. The integers in the input and output list are separated by a **,** (and nothing else).

Example 1

```
Input :  
DD  
1  
[42]
```

```
Output :  
error
```

Example 2

```
Input :  
RDD  
4  
[1, 2, 3, 4]
```

```
Output :  
[2, 1]
```