2019
Canadian Computing Competition
Senior Division

March 6, 2019

Problem S1: Filter the string

Time limit: 1.0 second(s)
Memory limit: 512 MB

Description

Given a string of lowercase letters a to z, your task is to delete all characters in the string that appear less than k times and output the generated string.

Input Format

Read from standard input.

The first line of the input contains one positive integers k.

The second line of the input contains a string S consisting of lowercase letters a to z. The length of S is no larger than 10^4.

Output Format

Write to the standard output.

Output only one string, representing the new generated string.

Sample Input 1

```
3
abcdaabccc
```

Sample Output 1

```
acaaccc
```

Sample Explanation 1

In this string, a appears 3 times, b appears twice, c appears 4 times, d appears once. So all the characters b and d in the string will be deleted.

Sample 2

See ***2.in*** and ***2.ans*** in the problem path.

Subtasks

There are 5 subtasks. The constraint for each subtask is shown as following:

Subtask	k	Length of S	Score
1	≤ 1	$\leq 10^2$	20
2	≤ 2		20
3	≤ 10		20
4	$\leq 10^3$	$\leq 10^3$	20
5	$\leq 10^4$	$\leq 10^5$	20

Problem S2: The square root

Time limit: 1.0 second(s)
Memory limit: 512 MB

Description

Given a perfect square number in binary representation, your task is to calculate its non-negative square root.

For example, we have $11001_{(2)} = 101_{(2)} \times 101_{(2)}$. If the input is `11011`, the output should be `101`.

Input Format

Read from standard input.

The input contains only one string with digits `0` and `1`, representing the perfect square number in binary.

Let n be its length, we promise $2 \leq n \leq 10^5$.

Output Format

Write to the standard output.

Output only one string with digits `0` and `1`, representing the non-negative square root of the given number in binary.

Sample Input 1

```
10010000
```

Sample Output 1

```
1100
```

Sample 2

See ***2.in*** and ***2.ans*** in the problem path.

Subtasks

There are 6 subtasks. The score for each subtask is the same. The constraint for each subtask is shown as following:

Subtask	n	Score
1	≤ 10	10
2	≤ 50	10
3	≤ 60	10
4	≤ 300	30
5	$\leq 3,000$	30
6	$\leq 10^5$	10

Hint

Due to the limited competition time, you'd better start to tackle the last subtask of this problem after finishing all the other problems, especially if your idea of the solution is not easy to code.

Problem S3: Integer circle

Time limit: 3.0 second(s)
Memory limit: 512 MB

Description

Given a circle consisting of n positive integers $a_0, a_1, \ldots, a_{n-1}$ (a_i and a_{i+1} are adjacent for $0 \leq i < n-1$, while a_0 and a_{n-1} are also adjacent), each in the range $[1, 40]$. You can take two adjacent numbers with equal values and replace them by a single number of value one greater (e.g., you might replace two adjacent 7 's with an 8). You can make the above moves as many times as you want. The goal is to maximize the value of the largest number present in the circle.

Now given $a_0, a_1, \ldots, a_{n-1}$ and n, your task is to calculate the maximum value that can be obtained.

Input Format

Read from standard input.
The first line of the input contains one positive integer n.
While the second line of the input contains n positive integers $a_0, a_1, \ldots, a_{n-1}$.
We have $n \leq 262,144$ and $a_0, a_1, \ldots, a_{n-1} < 40$.

Output Format

Write to the standard output.
Output only one integer which represents the maximum value that can be obtained.

Sample Input 1

```
6
3 4 4 3 3 3
```

Sample Output 1

```
6
```

Sample Explanation 1

First we replace two pairs of adjacent 3's with two 4's and get totally four of 4's(note that a_0 and a_{n-1} are adjacent).

Then we replace two pairs of adjacent 4's with two 5's and finally get 6. Clearly there is no way to get an integer larger than 6.

Sample 2

See ***2.in*** and ***2.ans*** in the problem path.

Subtasks

Your submitted code will be judged in subtasks. You must pass all the cases in one subtask to get the score of this subtask.

Subtask	n	Score
1	≤ 32	10
2	≤ 512	30
3	$\leq 4,096$	40
4	$\leq 262,144$	20

Problem S4: Summary, power and the square root

Time limit: 1.0 second(s)

Memory limit: 512 MB

Description

Given the following formula:

$$\left(\sqrt{2}+\sqrt{3}\right)^n+\left(\sqrt{2}-\sqrt{3}\right)^n+\left(-\sqrt{2}+\sqrt{3}\right)^n+\left(-\sqrt{2}-\sqrt{3}\right)^n.$$

We call $\left(\pm\sqrt{2}\pm\sqrt{3}\right)^n$ a term of this formula, and there are 4 terms in total.

For any positive integer n, none of the terms in this formula is an integer. However, it is interesting that the result of the entire formula, i.e., the sum of these terms, is an integer.

Generally, if $a_1, a_2, \ldots, a_m$ are positive integers, the result of the following formula

$$S_n=\sum\left(\pm\sqrt{a_1}\pm\sqrt{a_2}\pm\cdots\pm\sqrt{a_m}\right)^n$$

will also be an integer. The number of the terms in this formula is 2^m in total.

Now given $a_1, a_2, \ldots, a_m$ and n, your task is to calculate the result of $S_n \bmod 1,000,000,007$.

Input Format

Read from standard input.

The first line of the input contains two positive integers n and m, separated by one space.

The second line of the input contains m positive integers $a_1, a_2, \ldots, a_m$. Any two adjacent integers are separated by one space.

We have $m \le 7$ and $n, a_1, a_2, \ldots, a_m < 1,000,000,007$.

Output Format

Write to the standard output.

Output only one integer which represents $S_n \bmod 1,000,000,007$.

Sample Input 1

```
4 2
2 3
```

Sample Output 1

```
196
```

Sample Explanation 1

$$\left(\sqrt{2}\pm\sqrt{3}\right)^4 = 2^{\frac{4}{2}}\pm 4\times 2^{\frac{3}{2}}\times 3^{\frac{1}{2}}+6\times 2^{\frac{2}{2}}\times 3^{\frac{2}{2}}\pm 4\times 2^{\frac{1}{2}}\times 3^{\frac{3}{2}}+3^{\frac{4}{2}} = 4\pm 8\sqrt{6}+36\pm 12\sqrt{6}+9 = 49\pm 20\sqrt{6}$$

$$\left(-\sqrt{2}\pm\sqrt{3}\right)^4 = \left(\sqrt{2}\mp\sqrt{3}\right)^4 = 49\mp 20\sqrt{6}$$

$$S_4 = 49\times 4 = 196$$

Sample Input 2

```
18 3
3 7 18
```

Sample Output 2

```
336765963
```

Sample Input 3

```
988 4
207559639 532737791 634932890 335818535
```

Sample Output 3

```
754063451
```

Subtasks

There are 10 subtasks. The score for each subtask is the same. The constraints for each subtask is shown as following:

Subtask	n	m
1	$< 1,000,000,007$	1
2		2
3		3
4		4
5	$< 10^3$	5
6		6
7		7
8	$< 1,000,000,007$	5
9		6
10		7

Problem S5: Union on a tree

Time limit: 1.0 second(s)
Memory limit: 512 MB

Description

You are given an undirected connected graph of n vertices and $n-1$ weighted edges (i.e. an unrooted tree with weighted edges). The vertices are numbered from 1 to n. The j-th edge connects vertices u_j and v_j, of weight w_j. The distance between two vertices is defined as the total weight of the edges on the unique path between them.

You need to maintain multiple sets, each containing several vertices of the graph. Initially there are n sets, where the i-th set only contains the i-th vertex. Then you need to handle $n-1$ requests numbered from $n+1$ to $2n-1$. For request numbered i, you are given a_i and b_i, where $a_i < b_i < i$, and you need to assign the i-th set to be the union of the a_i-th and the b_i-th sets. It is guaranteed that for every set except set $2n-1$, it is unioned (as parameter a or b) exactly once. Thus set $2n-1$ must contain all the vertices.

Given an integer k, your task is to determine, for each set numbered from $n+1$ to $2n-1$, whether this set contains any 2 vertices between which the distance is at least k.

Input Format

Read from standard input.

The first line of the input contains 2 positive integers n and k.

The next $n-1$ lines contain information of the edges, where the j-th line contains 3 integers u_j, v_j and w_j. It is guaranteed the input specifies an undirected connected graph.

The next $n-1$ lines contain information about the requests, where the i-th line contains 2 integers a_{i+n} and b_{i+n}.

General constraints:

- $1 \le n \le 10^5$.
- $0 \le w_i \le 10^4$ for any i.
- $0 \le k \le 10^9$.

Output Format

Write to the standard output.

Output $n-1$ lines of integer 0 or 1. For i-th line, output 1 if set $i+n$ contains 2 vertices between which the distance is at least k, or output 0 otherwise.

Sample Input 1

```
5 3
3 1 2
3 2 1
3 4 1
4 5 1
1 5
2 3
4 7
6 8
```

Sample Output 1

```
1
0
0
1
```

Sample Explanation 1

Set 6 is $\{1, 5\}$, and the distance between vertices 1 and 5 is 4, which is at least $k = 3$.

Sample 2

See ***2.in*** and ***2.ans*** in the problem path.

Sample 3

See ***3.in*** and ***3.ans*** in the problem path.

Subtasks

Your submitted code will be judged in subtasks. You must pass all the cases in one subtask to get the score of this subtask.

Subtask	n	Extra constraints	Score
1	≤ 2		5
2	≤ 200		20
3	$\leq 2,000$		20
4	$\leq 3 \times 10^4$	$u_j = j, v_j = j + 1$ for any j	20
5	$\leq 3 \times 10^4$		20
6	$\leq 10^5$		15