

# Hong Kong Secondary Coding Challenge

City University of Hong Kong

August 6, 2022

- A Sequence
- B String Matching
- C GCD Ring
- D Businessman
- E 0-1 Ring

## Reminders (please read carefully)

1. The return value type of the main function in C++ must be int, and the return value when the program ends normally must be 0 .
2. If there is no special instructions, the comparison method of the results is full text comparison (ignoring the spaces at the end of the line and the empty line at the end of the text).
3. The submitted code should be no longer than 50 kilobytes.
4. The detailed compilation options are as follows.

C++	clang++/g++	-O2 -lm -DONLINE_JUDGE -mx32 -std=c++03
C++11	clang++/g++	-O2 -lm -DONLINE_JUDGE -mx32 -std=c++11
C++17	clang++/g++	-O2 -lm -DONLINE_JUDGE -mx32 -std=c++17
Pascal	fpc	-O2

## Problem A. Sequence (sequence.c/cpp/pas)

Input file: `stdin`  
Output file: `stdout`  
Time limit: 1 second  
Memory limit: 256 megabytes

Libra is keen on sequences. For a sequence, she thinks the smaller the sum of the digits of all numbers is, the better the sequence is.

Rahn sends Libra a sequence of length  $n$ ,  $a_1, a_2, \dots, a_n$ . However, Libra thinks the sequence is not good enough. So, Rahn allows Libra to perform some operations, each of which is to select a **positive integer** in the sequence and reduce it by 1. Libra can do at most  $k$  operations.

Please help Libra find the smallest sum of digits after doing the above operations.

### Input

The first line contains two integers  $n$  and  $k$ .

The following line contains  $n$  integers,  $a_1, a_2, \dots, a_n$ .

### Output

Output an integer in a line, representing the smallest sum of digits that can be obtained after no more than  $k$  operations.

### Example

stdin	stdout
3 6 114 514 1919810	39

### Notes

For all test data,  $1 \leq n \leq 5 \times 10^5, 1 \leq k \leq 10^9, 0 \leq a_i < 10^9$ .

Subtask	Score	Additional constraints
1	20	$n, k \leq 5$
2	20	$n, k \leq 5000$
3	20	$k \leq 5 \times 10^5$
4	40	No additional constraints

## Problem B. String Matching (match.c/cpp/pas)

Input file: `stdin`  
Output file: `stdout`  
Time limit: 1 second  
Memory limit: 256 megabytes

Given two strings  $S, T$ , where  $S$  only contains lowercase letters, and  $T$  contains lowercase letters and two types of matching symbols '.' and '\*'.

The same letters in  $S$  and  $T$  can match each other. The matching rules of '.' and '\*' are as follows.

1. '.' can be replaced by any lowercase letter.
2. '\*' can repeat the lowercase letter to the left of it any number of times (possibly zero).

For example, string ".a\*" can match "ba", "baaa", "zaaaa".

Please calculate how many **nonempty prefixes** of  $S$  can be matched by  $T$ .

### Input

The first line contains an integer  $n$ , representing the number of test cases.

Each test case consists of two lines, containing strings  $S$  and  $T$ .

### Output

For each test case, print one line containing the number of nonempty prefixes that can be matched by  $T$ .

### Example

stdin	stdout
5	5
aaaaa	5
a*	1
aaaaa	2
.*	2
abbbb	
.*	
aaabbb	
a*.b.*	
aaaaaaabcabc	
.*...abc	

### Notes

Denote the length of  $S$  (resp.  $T$ ) by  $|S|$  (resp.  $|T|$ ).

For all test data,  $1 \leq |S|, |T| \leq 2000, 1 \leq n \leq 5$ .  $S$  only contains lowercase letters, and  $T$  contains lowercase letters, '.' and '\*'. It is guaranteed that the first letter of  $T$  is not '\*'.

Subtask	Score	Additional constraints
1	20	$1 \leq  S ,  T  \leq 10$
2	20	$1 \leq  S ,  T  \leq 100$
3	20	$T$ does not contain '.'
4	40	No additional constraints

## Problem C. GCD Ring (gcd.c/cpp/pas)

Input file: `stdin`  
 Output file: `stdout`  
 Time limit: 1 second  
 Memory limit: 256 megabytes

There are  $n$  numbers  $a_1, a_2, \dots, a_n$  connected by  $n$  **different** ropes, forming a ring.

You can cut some of these ropes to divide the ring into several parts. Let  $s_1, s_2, \dots, s_k$  be the sum of numbers in each part, then the score of this cutting is  $\gcd(s_1, s_2, \dots, s_k)$ .

Let  $f_k$  be the **product** of scores among all ways to cut  $k$  ropes. Two cuttings are considered different if and only if the sets of the remaining ropes are different, namely, namely, there is at least one difference in positions to cut the ring between the two cuttings.

For all  $0 \leq k \leq n$ , calculate  $f_k$  module  $10^9 + 7$ .

### Input

The first line contains an integer  $n$ .

The second line contains  $n$  integers,  $a_1, a_2, \dots, a_n$ .

### Output

Output  $n + 1$  integers in a line, representing  $f_0, f_1, \dots, f_n$  module  $10^9 + 7$ .

### Example

stdin	stdout
3 1 2 3	6 216 6 1
6 2 2 2 2 2 2	12 2985984 56623104 4194304 32768 64 2

#### Explanation to the first example:

Denote the rope between  $a_i$  and  $a_{i+1}$  by  $(i)$  for  $1 \leq i \leq n - 1$ . Specially, we denote the rope between  $a_n$  and  $a_1$  by  $(n)$ . All possible cuts are as follows:

- Do nothing:  $\gcd(\{6\}) = 6$
- Cut (1):  $\gcd(\{6\}) = 6$
- Cut (2):  $\gcd(\{6\}) = 6$
- Cut (3):  $\gcd(\{6\}) = 6$
- Cut (1) and (2):  $\gcd(\{4, 2\}) = 2$
- Cut (1) and (3):  $\gcd(\{1, 5\}) = 1$

- Cut (2) and (3):  $\gcd(\{3, 3\}) = 3$
- Cut (1), (2) and (3):  $\gcd(\{1, 2, 3\}) = 1$

## Notes

For all test data,  $3 \leq n \leq 10^5, 1 \leq a_i \leq 10^9$ .

Subtask	Score	Additional constraints
1	5	$n \leq 20$
2	15	$n \leq 40, a_i \leq 10$ , and all $a_i$ 's are generated randomly
3	20	$n \leq 100$
4	30	$n \leq 2000$
5	30	No additional constraints

## Problem D. Businessman (businessman.c/cpp/pas)

Input file: `stdin`  
Output file: `stdout`  
Time limit: 2.5 second  
Memory limit: 256 megabytes

There are  $n$  countries connected by  $n - 1$  roads. It is guaranteed that there is exactly one simple path between any two countries.

As a businessman, you are in the first country and have  $x$  money initially.

You can earn  $a_i$  money in the  $i^{\text{th}}$  country (regardless of how many times you visit it). If you want to pass a road between two countries, you need to buy a road pass for this road, which costs some certain money (a road pass can be used any number of times on the corresponding road).

At any moment of your business travel, your money left cannot be less than 0. Please calculate the maximum money you can get (**including initial money**).

### Input

The first line contains two integers  $n$  and  $m$ , indicating the number of countries and number of queries.

The second line contains  $n$  integers  $a_1, a_2, \dots, a_n$ , indicating the money you can earn in each country.

Then  $n - 1$  lines follow. Each line contains three integers  $u, v$  and  $w$ , indicating a road between the  $u^{\text{th}}$  country and  $v^{\text{th}}$  country, and the pass check for this road costs  $w$  money.

In the next  $m$  lines, each line consists of a single integer  $x$ , representing a query. For this query, your initial money is  $x$ .

### Output

For each query, output an integer in a line, representing the maximum money you can earn.

## Example

stdin	stdout
5 5	2
1 2 3 4 5	9
1 2 3	10
1 3 4	11
2 4 1	12
2 5 1	
1	
2	
3	
4	
5	

## Notes

For all test data,  $1 \leq n \leq 10^6$ ,  $1 \leq m \leq 2 \times 10^5$ ,  $1 \leq a_i, w \leq 10^9$ ,  $0 \leq x \leq 10^{15}$ ,  $1 \leq u, v \leq n$ .

Subtask	Score	Additional constraints
1	5	$m = 1, n \leq 18$
2	10	$m = 1, n \leq 21$
3	15	$n \leq 100, m \leq 1000, a_i \leq 10, w \leq 10$
4	15	$u = i, v = i + 1, m \leq 10^5$
5	15	$n, m \leq 2000$
6	20	$n, m \leq 2 \times 10^5$
7	20	No additional constraints

## Problem E. 0-1 Ring (ring.c/cpp/pas)

Input file: `stdin`  
 Output file: `stdout`  
 Time limit: 2.5 second  
 Memory limit: 256 megabytes

For any ring consisting of  $n$  boolean values, we call it a good ring if there are no consecutive 0's with length larger than  $m$ .

Denote the number of different good rings with length  $n$  by  $f(n)$ . Two rings are considered different if one cannot be obtained by rotating the other ring. You are asked to calculate  $f(i)$  module 998244353 for all  $1 \leq i \leq N$ .

### Input

The input data contains two integers in a line,  $N, m$ , with the meaning shown in the description above.

### Output

You only need to print  $(f(1) \pmod{998244353}) \oplus (f(2) \pmod{998244353}) \oplus \dots \oplus (f(N) \pmod{998244353})$ , where  $\oplus$  stands for the bitwise-xor operator.

### Example

stdin	stdout
4 3	0
8 2	18
30 15	64891815
130 55	824979376

### Notes

For all test data,  $1 \leq m \leq N \leq 2 \times 10^7$ .

Subtask	Score	Additional constraints
1	20	$N \leq 18$
2	30	$N \leq 10^3$
3	20	$N \leq 10^6$
4	30	No additional constraints