

**The 2025 China-ASEAN International Collegiate
Programming Contest &
The 8th Guangxi Collegiate Programming Contest
Contest Session
May 25, 2025
Problem List**

A	Additive Combinatorics
B	Beats
C	Call for Papers
D	DeepSeek, DeepThink!
E	Elimination
F	Faster Elimination
G	Grand XOR Counting Problem Challenge II
H	Hollow Knight: Silksong
I	Identify the Feasibility
J	Join the Joint
K	Kingdom Rush 5: Alliance
L	Lost in MST

**Do not open the problem set before the competition begins!
This problem set should contain 12 problems on 16 numbered
pages. Please inform a volunteer immediately if something is
missing from your problem set.**

Prepared by Hangzhou Dianzi University

Problem A. Additive Combinatorics

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 256 megabytes

Recently, Colin became addicted to the theory of additive combinatorics, which is about counting additive structures in sets. This theory has seen exciting developments and dramatic changes in direction in recent years, thanks to its connections with areas such as number theory, ergodic theory, and graph theory.

One fundamental problem in additive combinatorics is **sumsets**. For two sets of integers A, B , we define the sumset of A and B to be the set of all sums of an element from A with an element from B . That is,

$$A + B = \{a + b : a \in A, b \in B\}$$

Now Colin gives you three sets A, B, C ; could you determine whether C is the sumset of A and B ?

Input

The first line contains three integers n_1, n_2, n_3 ($1 \leq n_1, n_2 \leq 10^3, 1 \leq n_3 \leq 10^6$), representing the sizes of A, B , and C .

The second line contains n_1 integers $a_1, a_2, \dots, a_{n_1}$ ($1 \leq a_i \leq 10^9$), representing the numbers in A .

The third line contains n_2 integers $b_1, b_2, \dots, b_{n_2}$ ($1 \leq b_i \leq 10^9$), representing the numbers in B .

The fourth line contains n_3 integers $c_1, c_2, \dots, c_{n_3}$ ($1 \leq c_i \leq 2 \times 10^9$), representing the numbers in C .

It is guaranteed that the numbers in the same set are pairwise distinct.

Output

If C is the sumset of A and B , output YES. Otherwise, output NO.

Examples

standard input	standard output
2 2 3 1 2 2 1 3 4 2	YES
2 2 2 1 2 1 2 2 3	NO
2 2 4 1 2 1 2 1 2 3 4	NO

Problem B. Metronome for Corgi

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

Azora the corgi loves music but cannot catch the rhythm well, especially when counting beats! After hard research on music theory, Azora found a hidden theorem: There won't be a note crossing the boundary between the beats for modern songs (or for most barking sessions for that matter).



To simplify the problem, we assume that a modern song consists of several sequential notes whose lengths are $l_1, l_2, \dots, l_n$ milliseconds. The notes will be played one by one, which means the song will start playing the first note at time 0, and the second note at the beginning of the $l_1 + 1$ milliseconds, and the third note at the beginning of the $l_1 + l_2 + 1$ milliseconds, and so on.

If the length of the beat is L milliseconds, then there should not exist any note that is playing but not starting to play at the beginning of the kL milliseconds, for any non-negative integer k . Formally, there should not exist any note i such that $\sum_{j=1}^{i-1} l_j < kL$ but $\sum_{j=1}^i l_j > kL$ for any non-negative integer k .

Now Azora has received a new song from her humans, and she wants to practice counting beats before she barks the song back to them. To challenge herself, Azora wants to find the smallest possible length of a beat, such that it satisfies the theorem. Can you help her?

Input

The first line contains a single integer n ($1 \leq n \leq 2 \times 10^5$), representing the number of notes in the song. The second line contains n integers $l_1, l_2, \dots, l_n$ ($1 \leq l_i \leq 10^9$), representing the length of each note.

Output

Output a single integer, representing the minimum possible length of a beat.

Examples

standard input	standard output
5 1 2 3 6 5	6
5 1 2 3 4 5	10

Problem C. Call for Papers

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Colin is a first-year Ph.D. student, working hard with many beautiful but unrealistic academic fantasies. He aims to design clever algorithms for essential problems and publish them in top conferences.

There are n conferences Colin is interested in. The i -th one starts receiving submissions at 8 a.m. on the day r_i , and the deadline to submit papers is 8 p.m. on the day d_i . After the submitted paper has been reviewed by the experts, the authors will be notified of the result at 2 p.m. on the day t_i . The result is either **Accepted** or **Rejected**. In addition, there is an impact factor w_i for the i -th conference, which represents the reputation of the conference. To simplify the problem, we guarantee that the impact factors of conferences are **pairwise distinct**.

But recently, he sadly found that his papers were rejected again and again. By summarizing the past data, Colin raised an assumption: the probability of acceptance for a paper is equal to the number of times it has been submitted over the impact factor of the conference. Formally, if Colin submits a paper to the i -th conference now, and it has already been submitted for k times (including this time), then the probability of acceptance this time is equal to

$$\min(1, \frac{k}{w_i})$$

Now Colin has m papers he would like to submit, and the i -th paper will be finished at 9 a.m. on the day f_i . We say a conference is available at a time if it has already started receiving papers, but its deadline hasn't passed. Once Colin finishes each paper, if there does not exist any available conference at that time, then he will think this paper is not welcome and won't submit it anymore; if there exists any available conference at that time, he will instantly submit it to one of the available conferences with the highest impact factor. Then Colin will repeat as follows:

On the day of receiving the notification of the result, if the paper is **Rejected**:

- If any available conference exists **at that time**, Colin will instantly submit the paper to one of the available conferences with the highest impact factor and wait for the result.
- Otherwise, Colin will be pretty upset about this paper and won't submit it anymore, even if there are some further conferences that haven't started receiving papers.

If the paper is **Accepted**, Colin will earn reputation points, which are equal to the impact factor of the conference that accepts the paper. He will not submit the paper to other conferences.

Colin would like to ask, if his assumption is valid, what is the expected value of reputation points he can earn from each paper? If a paper is not accepted in the end, then Colin will earn 0 reputation points from it. Please find the answer modulo 998244353.

Notice that all the times mentioned are in AOE time, and there is no need to consider any issues related to time zones.

Input

The first line contains two integers n, m ($1 \leq n \leq 5 \times 10^3$, $1 \leq m \leq 10^5$), representing the number of conferences Colin is interested in, and the number of papers Colin would like to submit.

For the following n lines, each line contains four integers r_i, d_i, t_i, w_i ($1 \leq r_i < d_i < t_i \leq 10^9$, $1 \leq w_i \leq 10^8$), representing the information for the i -th conference: it starts receiving submissions from 8 a.m. on day r_i , the deadline is 8 p.m. on day d_i , and authors will be notified of the results at 2 p.m. on day t_i whether their submissions are accepted. It's guaranteed that all the w_i are pairwise distinct.

For the following m lines, each line contains a single integer f_i ($1 \leq f_i \leq 10^9$), representing that Colin will finish the i -th paper at 9 a.m. on the day f_i .

Output

For each paper, print a line with a single integer, representing the expected value of reputation points Colin could earn from it, modulo 998244353.

Example

standard input	standard output
5 3	1
1 2 3 1	249561091
2 3 4 2	332748120
3 4 5 3	
4 5 6 4	
5 6 7 5	
1	
2	
3	

Note

For the first paper, Colin will submit it to the first conference, and according to the assumption, the paper will definitely be accepted. Thus, the expected value of reputation he could earn is equal to the impact factor of the first conference, which is 1.

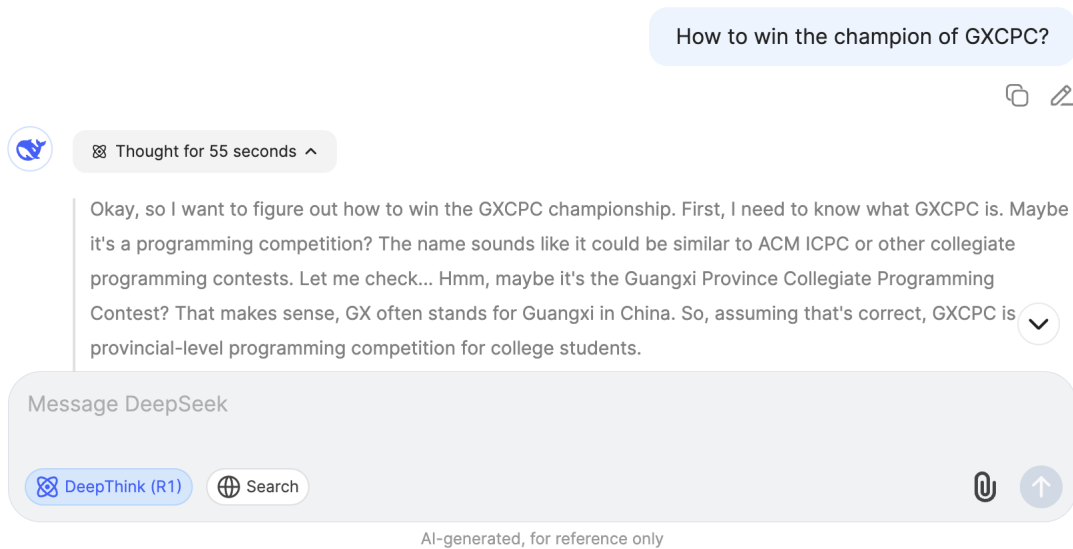
For the second paper, Colin will first submit it to the second conference with 50% probability of being accepted. If it is rejected, he will receive the notification on the day 4, then he will submit it to the fourth conference, with $\frac{2}{4} = 50\%$ probability of being accepted. If it is still rejected, then he will receive the notification on day 6, then he will submit it to the fifth conference, with $\frac{3}{5} = 60\%$ probability of being accepted. If it is still rejected, then there is no more chance to submit it. Thus, the expected value is $2 \times 50\% + 4 \times 50\% \times 50\% + 5 \times 50\% \times 50\% \times 60\% = \frac{11}{4}$, which equals 249561091 modulo 998244353.

Problem D. DeepSeek, DeepThink!

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Recently, Colin found that DeepSeek is such a useful AI that he can always give it problems he meets and receive lots of good suggestions, especially when Colin turns on the DeepThink option: DeepSeek will show the procedure of thinking, and it makes everything look so clear and convincing!

Winning Strategies for GX CPC Success



Although DeepSeek's thinking procedure does look like a human's, Colin found a phenomenon that always starts with, "Okay, so I want to figure out how to..."

Colin wants to build an extraordinary AI by himself: ColinSeek, but since Colin knows too little about AI, he decides to start by imitating. Now, please help Colin write the very first code for the ColinThink option of ColinSeek:

When the user asks in the format **How to xxx?**, show the thinking procedure as **Okay, so I want to figure out how to xxx**. The last part (xxx) is the same as the user's question.

Input

A single line represents the user's question. The input is guaranteed to start with "How to " and end with "?". The input between these two parts consists of lower English letters, digits, and blank spaces. Except for the two sample inputs, the length of the input line (excluding the line ending) is larger than 1024 bytes and less than 16384 bytes.

Output

Output the thinking procedure of ColinSeek, as described in the description.

Examples

standard input	standard output
How to win the champion of gxcpc?	Okay, so I want to figure out how to win the champion of gxcpc.
How to solve these 13 questions ?	Okay, so I want to figure out how to solve these 13 questions .

Note

There is a line break in the output of the samples in the PDF, just for better demonstration. Please output in a single line in your submission, and the outputs of the samples shown on the platform are correct.

Problem E. Elimination

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

This is the easy version of the problem. The only difference between the versions is the range of n .

Colin likes to play the Happy Eliminate game to kill time. Usually, the player is asked to find a fixed pattern in the given content, then eliminate it. The variety of patterns makes the game rich and exciting.

To research which pattern makes the game most exciting, Colin built a model: The contents given to the user are represented as a string s , and the pattern is another string t . In each turn of the game, the user can do one of the following two operations, and the game will be kept going until t becomes empty:

- Delete the last character of the pattern t .
- Find a substring of s that is equal to t , say $s[i, j] = t$, then delete several characters in the prefix of s until $s[i + 1]$, i.e., (make s become $s[i + 1, |s|]$), and then delete the last character of the pattern t .

Colin defines a game's excitement as the maximum number of times the second operation can be performed. He wants to find the best pattern to maximize excitement, but he has no idea how to construct it. Thus, he turned it over to you, the top player of Happy Eliminate.

Recall that $|s|$ represents the length of the string s , and $s[i, j]$ represents the substring of the string s , starting from the i -th character to the j -th character (including the i -th and the j -th).

Input

The first line contains a single integer T , representing the number of test cases.

For each test case:

The first line contains a single integer n ($1 \leq n \leq 6000$), representing the length of the string s .

The second line contains a string s , consisting only of lowercase letters.

It's guaranteed that the sum of n is not larger than 6000.

Output

For each test case, output a single line with a single integer, representing the maximum possible excitement.

Example

standard input	standard output
3	3
9	4
abcabcabc	3
4	
aaaa	
5	
ababa	

Note

For the first sample, you can choose t as abc .

For the second sample, you can choose t as $aaaa$.

For the third sample, you can choose t as aba .

Problem F. Faster Elimination

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

This is the hard version of the problem. The only difference between the versions is the range of n .

Colin likes to play the Happy Eliminate game to kill time. Usually, the player is asked to find a fixed pattern in the given content, then eliminate it. The variety of patterns makes the game rich and exciting.

To research which pattern makes the game most exciting, Colin built a model: The contents given to the user are represented as a string s , and the pattern is another string t . In each turn of the game, the user can do one of the following two operations, and the game will be kept going until t becomes empty:

- Delete the last character of the pattern t .
- Find a substring of s that is equal to t , say $s[i, j] = t$, then delete several characters in the prefix of s until $s[i + 1]$, i.e., (make s become $s[i + 1, |s|]$), and then delete the last character of the pattern t .

Colin defines a game's excitement as the maximum number of times the second operation can be performed. He wants to find the best pattern to maximize excitement, but he has no idea how to construct it. Thus, he turned it over to you, the top player of Happy Eliminate.

Recall that $|s|$ represents the length of the string s , and $s[i, j]$ represents the substring of the string s , starting from the i -th character to the j -th character (including the i -th and the j -th).

Input

The first line contains a single integer T , representing the number of test cases.

For each test case:

The first line contains a single integer n ($1 \leq n \leq 5 \times 10^5$), representing the length of the string s .

The second line contains a string s , consisting only of lowercase letters.

It's guaranteed that the sum of n is not larger than 5×10^5 .

Output

For each test case, output a single line with a single integer, representing the maximum possible excitement.

Example

standard input	standard output
3	3
9	4
abcabcabc	3
4	
aaaa	
5	
ababa	

Note

For the first sample, you can choose t as abc .

For the second sample, you can choose t as $aaaa$.

For the third sample, you can choose t as aba .

Problem G. Grand XOR Counting Problem Challenge II

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 1024 megabytes

The annual GXPCP (Grand XOR Counting Problem Challenge) is here again! This year, Judge Colin has prepared another super difficult counting problem for the contestants! Can you slay this ultimate counting problem and win the championship?

Eva has two magic parameters k_1 and k_2 .

Colin has two integer sequences $a_1, a_2, \dots, a_n$ and $b_1, b_2, \dots, b_n$.

They want to know how many pairs of indices (i, j) satisfy:

- $1 \leq i < j \leq n$
- $k_1 \oplus a_i \oplus a_j < k_2 \oplus b_i \oplus b_j$

Where $\oplus$ represents the bitwise XOR operation.

Input

The first line contains three integers n, k_1, k_2 ($1 \leq n \leq 2 \times 10^5, 1 \leq k_1, k_2 < 2^{30}$), representing the length of the two sequences and the magic parameters.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i < 2^{30}$), representing the first sequence.

The third line contains n integers $b_1, b_2, \dots, b_n$ ($1 \leq b_i < 2^{30}$), representing the second sequence.

Output

A single integer, representing the number of pairs satisfying the condition.

Example

standard input	standard output
5 1 2 3 4 5 6 7 1 2 3 1 2	2

Note

There are two valid pairs in the sample: $(2, 3)$ and $(4, 5)$

Problem H. Hollow Knight: Silksong

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

Colin is a big fan of Hollow Knight. The grand world view of Hollownest and the distinctive bosses deeply attracted him, and fortunately, the second generation, Hollow Knight: Silksong, will be released very soon! Silksong follows Hornet, the former princess of Hollow Knight's fallen kingdom of Hollownest and a supporting character in that game, as she is forced to explore a totally new land.



So many people are excited to play Silksong! There's even a comment in 2020 on bilibili that has become famous: "Actually, I'm pretty lucky. I just got into Hollow Knight 1, and Hollow Knight 2 is about to come out." It has received over 28,000 likes now.



其实我蛮幸运的^o，刚入空洞一，快玩完了空洞二^o就要出了。

2020-04-07 01:25 👍 28398 💬 回复

Anyway, the most recent news is that Silksong will be released this year! The most exciting news is that people will have the chance to play the demo in Australia on **September 18, 2025**.

It cannot be luckier for those who haven't played Hollow Knight ever! So Colin strongly suggested that Eva play Hollow Knight before the second generation was released. To value how lucky Eva is, Colin defines the luckiness to be the number of dates between the first day Eva starts to play and the day of demo-play.

Given the date Eva starts to play Hollow Knight, can you help Colin calculate how lucky Eva is?

Input

A single line with three integers separated by a space y, m, d ($2017 \leq y \leq 2025, 1 \leq m \leq 12, 1 \leq d \leq 31$), representing the date Eva starts to play Hollow Knight in the year y , month m , and day d . It's guaranteed that the input date is valid, and it is earlier than the day of demo-play of Silksong (**September 18, 2025**) and later than the day Hollow Knight was released (**February 24, 2017**).

Output

A single integer, representing the luckiness of Eva.

Example

standard input	standard output
2025 5 25	116

Problem I. Identify the Feasibility

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 256 megabytes

Recently, Colin has been learning the relationship between sets and bitmasks. We can use binary numbers (bitmasks) to represent sets, and if some bits are 1 in the bitmask, then it means that the sets include the elements corresponding to the indices of these bits.

Now Colin has n non-negative integers $a_1, a_2, \dots, a_n$ and all these integers are less than 2^k . Now Colin tries to use the concept of bitmasks and sets to understand the relationship between these integers.

For a non-negative integer $x \in [0, 2^k)$, Colin defines that a_i is feasible with a_j under the augmentation of x , if and only if the set corresponding to the bitmask $a_i + x$ is a subset of the set corresponding to the bitmask $a_j + x$. Then Colin uses $f(x)$ to represent the number of all the pairs (i, j) such that a_i is feasible with a_j under the augmentation of x .

Now Colin would like to ask you to calculate the sum of all the $f(x)$, modulo 998244353 .

Formally, please calculate

$$\left(\sum_{x=0}^{2^k-1} f(x) \right) \bmod 998244353$$

Input

The first line contains two integers n, k ($1 \leq n \leq 10^6, 1 \leq k \leq 20$), representing the number of integers Colin has and all of these integers are less than 2^k .

The following line contains n integers $a_1, a_2, \dots, a_n$ ($0 \leq a_i < 2^k$).

Output

Output a single integer, representing the answer.

Example

standard input	standard output
2 2 2 0	10

Note

$f(0) = 3 : (1, 1), (2, 1), (2, 2)$

$f(1) = 3 : (1, 1), (2, 1), (2, 2)$

$f(2) = 2 : (1, 1), (2, 2)$

$f(3) = 2 : (1, 1), (2, 2)$

Problem J. Join the Joint

Input file: **standard input**
Output file: **standard output**
Time limit: **3 seconds**
Memory limit: **256 megabytes**

Recently, Eva learned several advanced data structure techniques for trees. To show her progress, she asked Colin to give her a challenging problem about tree statistics.

However, Colin just gave her a very typical problem:

Given a tree with n vertices, and vertex 1 is the root. Each vertex has an integer weight w_i .

Let $lca(u, v)$ represent the lowest common ancestor of the vertices u and v .

Given two integer sequences $a_1, a_2, \dots, a_x$ ($1 \leq a_i \leq n$) and $b_1, b_2, \dots, b_y$ ($1 \leq b_i \leq n$). Define the joint power of these two sequences to be

$$\sum_{i=1}^x \sum_{j=1}^y w_{lca(a_i, b_j)}$$

Please calculate the value of this joint power.

Eva thinks this problem is too easy, so she decided to change the problem into an online problem:

First, given the tree and the weight of each vertex. Initially, both sequences are empty.

Then there will be m operations, each one is of the following two types:

- 1 u : append u ($1 \leq u \leq n$) to the end of the first sequence.
- 2 u : append u ($1 \leq u \leq n$) to the end of the second sequence.

After each operation, you need to output the current joint power of the two sequences.

However, this version of the problem is too hard for Eva, so she added an additional limitation that, after each operation, the sum of values in the first sequence will not exceed 10^6 . Now Eva turns this problem to you; do you know how to solve it?

Input

The first line contains two integers n, m ($1 \leq n, m \leq 10^5$), representing the number of vertices in the tree and the number of operations.

The second line contains n integers $w_1, w_2, \dots, w_n$ ($1 \leq w_i \leq 10^5$), representing the weight of each vertex.

For the following $n - 1$ lines, each line contains two integers u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$), representing that there is an edge connecting u_i and v_i in the tree. It's guaranteed that all the edges in the input form a tree.

For the following m lines, each line contains two integers op_i, u_i ($op_i \in \{1, 2\}, 1 \leq u_i \leq n$), representing an operation. It's guaranteed that after each operation, the sum of values in the first sequence will not exceed 10^6 .

Output

For each operation, output a single line with a single integer, representing the joint power of the two sequences after the corresponding operation.

Examples

standard input	standard output
3 5 1 2 3 1 2 1 3 1 1 2 1 1 2 2 2 1 3	0 1 2 5 7
10 6 1 2 10 5 5 6 5 6 2 2 1 2 2 3 1 4 4 5 2 6 3 7 6 8 2 9 6 10 2 10 1 7 2 5 1 9 1 5 2 6	0 2 3 6 12 17

Problem K. Kingdom Rush 5: Alliance

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

Colin is a huge fan of the Kingdom Rush franchise tower defense games. Last year, KR released the latest generation of Kingdom Rush: *Kingdom Rush 5: Alliance*. As a formidable evil arises in the new generation, an unexpected alliance is taking shape! It will take the finest warriors of Linirea and the relentless Dark Army together to stop it. Therefore, the most attractive feature of the latest generation of games is that you can use two heroes simultaneously!



To measure the power of the double-hero feature, Colin would like to do some simple experiments: Assume that a hero's attack range is a circle on the 2D plane with its coordinates as the center. The path of enemies is a segment on the 2D plane. Colin would like to ask you how long the total length of the enemy's path is that is covered by at least one hero's attack range?

Input

The first line contains three integers x_1, y_1, r_1 ($|x_1|, |y_1| \leq 10^9, 1 \leq r_1 \leq 10^9$), representing the coordinates of the first hero as (x_1, y_1) , and the radius of its attack range is r_1 .

The second line contains three integers x_2, y_2, r_2 ($|x_2|, |y_2| \leq 10^9, 1 \leq r_2 \leq 10^9$), representing the coordinates of the second hero as (x_2, y_2) , and the radius of its attack range is r_2 .

The third line contains four integers a, b, c, d ($|a|, |b|, |c|, |d| \leq 10^9$), representing the path of enemies as a segment from (a, b) to (c, d) .

Output

A single line with a number, representing the total length of the enemy's path that is covered by at least one hero's attack range.

Your output will be considered correct if the absolute or relative error to the jury does not exceed 10^{-6} .

Examples

standard input	standard output
0 0 1 1 1 1 0 0 1 1	1.414213562373095
0 0 1 2 2 1 -1 -1 3 3	4.000000000000000

Problem L. Lost in MST

Input file: `standard input`
Output file: `standard output`
Time limit: 5 seconds
Memory limit: 1024 megabytes

One day, Colin got lost in a complete undirected graph G with n vertices, and each vertex i is associated with a magic parameter c_i . For each pair of vertices (i, j) , there exists an undirected edge connecting them with weight $w_{i,j} = \gcd(c_i, c_j) + \text{lcm}(c_i, c_j)$.

Can you find the graph's minimum spanning tree G to save Colin?

Recall that $\gcd(x, y)$ is the greatest common divisor of x and y , and $\text{lcm}(x, y)$ is the least common multiple of x and y .

Input

The first line contains a single integer n ($1 \leq n \leq 10^5$), representing the number of vertices in the graph.

The second line contains n positive integers $c_1, c_2, \dots, c_n$ ($1 \leq c_i \leq 10^6$).

Output

Output a single line with a single integer, representing the total weight of the edges in the minimum spanning tree of G .

Example

standard input	standard output
4 7 8 9 10	163