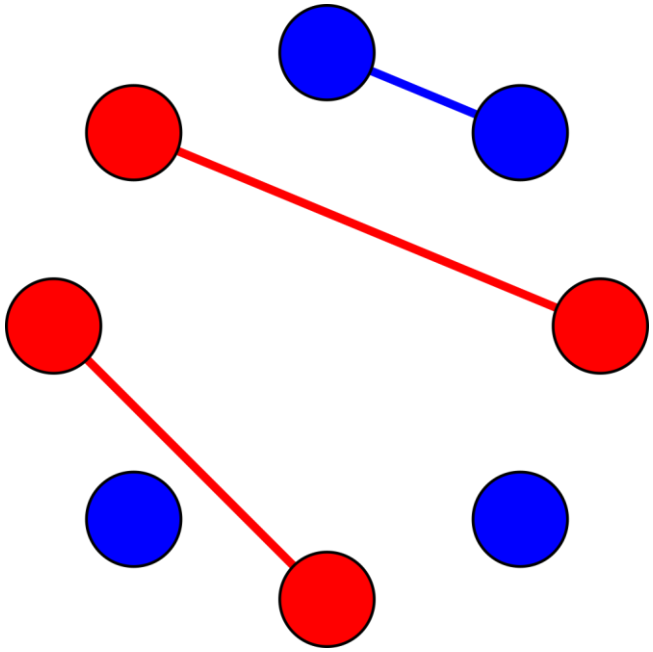
Problem A: Don't Cross the Beams

Filename: `beams`
Timelimit: `2 seconds`

Lasers are great fun! That is why you have setup a configuration of posts placed in a circular configuration with equal spacing. On each post you will place either a laser or a mirror box to receiver the laser beam!

You haven't quite figured out a configuration that looks nice but you do know for each post if you want it to send or receive laser beams of the two types of colors available (red or blue). You also don't want any of the beams to cross in your laser display.

Now you want to know the maximum number of lasers you can have in your design given a starting configuration of red and blue posts. Below is an example of posts and lasers being sent between them. Note that lasers travel in a straight line from their source to their mirror box.



Input

The input is a single string, ***s*** ($1 \le |s| \le 10^5$), of the characters 'R' or 'B', representing blue or red posts if you walked around the circle starting at an arbitrary post.

Output

Output a single integer on a line by itself representing the number of posts you can pair of the same color and not cross the laser beams!

Samples

Input	Output
RRRR	2
BBRBBRBRB	4
BRBRB	2
BBBRRRBBBRRR	5

Problem B: Excellence

Filename: `excellence`
Time limit: `2 seconds`

The World Coding Federation is setting up a huge online programming tournament of teams comprised of pairs of programmers. Judge David is in charge of putting teams together from the South Eastern delegation. Luckily, he has an even number of students who desire to compete, so that he can make sure that each student does compete. However, he'd like to maintain his pristine reputation amongst other judges by making sure that each of the teams he fields for the competition meet some minimum total rating. We define the total rating of a team to be the sums of the ratings of both individuals on the team. Help David determine the maximal value, X, such that he can form teams, each of which have a total rating greater than or equal to X. Note that every student must be placed on exactly one team of two students.

Input

Each input will consist of a single test case. Note that your program may be run multiple times on different inputs. The first line of each test case will be a positive even integer, n ($n \leq 10^5$), representing the number of students who desire to enter the online programming tournament. The following n lines will contain one single integer, s_i ($1 \leq s_i \leq 10^6$), each, representing the rating of student i.

Output

Output a single integer on a line by itself representing the maximal value, x, such that David can form teams where every team has a total rating greater than or equal to x.

Samples

Input	Output
4 1 2 3 5	5
2 18 16	34
4 13 12 19 14	27

Problem C: Sorting K Window Sums

Filename: `ksums`

Time limit: `2 seconds`

Given a sequence of ***n*** integers, we define a k-window to be k contiguous terms in the sequence. There are ***n*** - ***k*** + 1 k-windows in a sequence of ***n*** integers so long as ***n*** ≥ ***k***. We can label each of these windows with their starting index, with the leftmost window having the label 1. Naturally, we can define the sum of a specific k-window to be the sum of the terms within the window. We can sort each of the ***n*** - ***k*** + 1 k-windows in a sequence of ***n*** integers based upon the sum of each window, with windows with larger sums coming first. If two k-windows have the same sum, we break ties by having the one with the lower label come first.

For this problem, given a sequence of ***n*** integers and a value of ***k*,** determine the sorted order of each of the k-windows of the sequence.

Input

The first line will contain two space separated integers, ***n*** ($1 \le n \le 10^5$) and ***k*** ($1 \le k \le n$), representing the length of the input sequence and the size of the windows, respectively. The next line will contain ***n*** space separated integers x_i ($1 \le x_i \le 10^9$), representing the i^{th} value in the sequence.

Output

Output ***n*** - ***k*** + 1 space separated integers representing the sorted order (as defined above) of all of the k-windows, where each integer is the label of the window represented.

Samples

Input	Output
5 3 2 3 4 3 2	2 1 3
10 4 14 12 18 19 13 12 14 14 10 20	1 2 3 4 7 5 6

Problem D: One Dimensional Sovereigns

Filename: `one`

Time limit: `2 seconds`

The Count has ruled over this portion of the number line for long enough; it is time to appoint a pair of unique integers as king and queen to take over. The Count knows that the compatibility of the king and queen is vital to success, so he wants to pick a pair with the largest possible greatest common divisor. However, he also wants to pick the pair to be as small as possible to keep them humble.

Input

Each test case consists of a single line containing two space separated integers ***a*** and ***b*** ($1 \le \boldsymbol{a} < \boldsymbol{b} \le 5 \cdot 10^5$)

Output

Output a single line containing a pair integers x and y, ($\boldsymbol{a} \le x < y \le \boldsymbol{b}$) such that there is no other pair of integers inside the same range with a larger gcd. If there are multiple pairs with the same gcd, pick the pair with the smallest x. If there are still ties, pick the pair with the smallest y.

Samples

Input	Output
1 5	2 4
3 5	3 4
13 18	15 18
14 27	18 27
12 17	12 16
333126 333456	333141 333434

Problem E: Super Consulting

Filename: `consulting`
Timelimit: `3 seconds`

Brian Mulch is too smart for his own good. Companies hire him because of his reputation to give business advice, but they never believe him when he tells them that their entire business strategy is flawed. Convincing them is such a chore. Not only will employees not believe him unless their direct superiors agree with him, but he was to convince everyone personally. You would think that if all your superiors were convinced that would be convincing enough, but it appears that everyone else has trouble communicating even the simplest of ideas. Because he is so perceptive, Mr. Mulch knows exactly how much mental energy is necessary to convince someone and how much mental energy will be drained in the process. For any given firm Brian would like to know the minimum starting mental energy required to convince everyone. There may be multiple possible orderings in which to convince people and some may be more efficient than others.

Input

The first line of input contains a single positive integer n ($n \leq 20$), the number of employees at a firm. Following this are n lines, each with three space separated integers: s_i ($0 \leq s_i < n$, $s_i < i$), r_i ($0 \leq r_i \leq 10^3$), and d_i ($0 \leq d_i \leq r_i$), the supervisor of employee i or -1 if this employee is the CEO, the mental energy necessary to convince employee i, and the mental energy drained in the process, respectively. Employees are numbered starting at 0 in the order of the input and the CEO is always the employee numbered 0.

Output

On a line by itself, output a single integer, the minimum mental energy needed to convince everyone.

Samples

Input	Output
4 -1 2 0 0 2 1 0 5 2 1 10 1	11
3 -1 4 2 0 6 2 0 5 3	9

Problem F: Product of Digits

Filename: `product`
Timelimit: `4 seconds`

Let P(x) be a function that returns the product of digits of x. For example P(935) = 9*3*5 = 135.
Now let us define another function $P_{repeat}(x)$:

```
int P_repeat(int x){
      if(x < 10) return x
      return P_repeat(P(x))
}
```

Using 935 as an example once more, $P_{repeat}(935) = P_{repeat}(135) = P_{repeat}(15) = P_{repeat}(5) = 5$. Given a range from ***a*** to ***b*** and a target value ***v*** count all numbers x such that $P_{repeat}(x) = \boldsymbol{v}$ with $\boldsymbol{a} \leq x \leq \boldsymbol{b}$.

Input

The input has a single line with 3 space separated integers: ***a*** ($1 \leq \boldsymbol{a} \leq 10^{13}$), ***b*** ($\boldsymbol{a} \leq \boldsymbol{b} \leq 10^{13}$), and ***v*** ($0 \leq \boldsymbol{v} \leq 9$).

Output

On a line by itself, output a single integer *k,* the count of all numbers x in between a and b inclusive, such that $P_{repeat}(x) = \boldsymbol{v}$.

Samples

Input	Output
935 953 5	2
1 20 2	2
50 59 0	7
1 1000 7	6
33333 99999 1	0
1 10000000000000 9	455

Problem G: Video Game Story

Filename: `story`
Timelimit: `3 seconds`

Webros has finished what is now one of her favorite video games. What she notices about this particular video game is that there are several ways to finish the game and arrive at the ending of the game.

This game consists of n events indexed 1 through n, and event indexed i ($1 \leq i < n$) is followed by event e_i. Each event can only occur after all prior events leading up to it have already occurred.

Now Webros wonders how many unique ways (a way, in this context, is an ordering of the completion of the levels) she can play the game. Two ways are considered different if there are two events *a* and *b* where *a* occurs before *b* in one way, but *b* occurs before *a* in the other way.

Input

The first line of input contains single positive integer n ($2 \leq n \leq 5{,}000$), representing the number of events for the input case. The second line contains n - 1 space separated positive integers. The ith integer on this line, e_i, denotes that event i is required in order for event e_i to occur. The last event is guaranteed to require all of the first n - 1 events, directly or indirectly, and it's guaranteed that there exists at least one way to arrive at event n, satisfying all of the constraints.

Output

On a line by itself, output the *remainder* of the number of ways in which Webros can play the game when divided by 10^9+7.

Samples

Input	Output
4 2 3 4	1
5 2 5 4 5	6
7 2 4 4 7 7 5	45

Problem H: Cyber Security War

Filename: `war`

Time limit: `1 second`

Arup is running his cryptography class, CIS 3362 and has a great idea for a project. He wants to split the class into two teams and pit them against each other in a cyber-security contest. Unfortunately, several pairs of students are sworn enemies, so he can't put them on the same team. Since this is a team contest, he also requires that each team have at least two people. Help Arup determine whether or not he can split up his class into two teams.

Input

The first line of input will contain two space separated integers: s ($4 \leq s \leq 100$) and e ($0 \leq e \leq 2500$), representing the number of students in the class and the number of pairs of students who are enemies in the class, respectively. The students in the class are numbered from 0 to s-1. The following e lines will contain a pair of distinct space-separated integers representing two students in the class that are sworn enemies, who you may not place on the same team. Each pair listed will be unique.

Output

Output "YES" on a line by itself it there exists a way to split the class into two teams with each team having at least two people and "NO" otherwise.

Samples

Input	Output
4 5 0 1 0 2 1 2 1 3 0 3	NO
4 4 0 2 0 3 1 2 1 3	YES