Problem A
Animal Crossing

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Adam is a 10-year-old boy. He knows how to perform arithmetic operations which are addition, subtraction, multiplication, and division. Today, his older brother Alan gives him a challenge.Alan gives Adam two numbers p and q where $-1000 \leq p \leq q \leq 1000$.Alan wants Adam to compute the sum of the integer between p and q, inclusively. In other words, Adam is asked to compute $p + (p+1) + \cdots + q$. Adam does not want to perform so many arithmetic operations, since he wants to play "Animal Crossing." Please write a program to help Adam, because you want to play "Animal Crossing" with him.

Input Format

The input has exactly one line. That line contains exactly two integers p and q separated by a blank.

Output Format

Output the answer to Alan's challenge on one line.

Technical Specification

- $-1000 \leq p \leq q \leq 1000$

Sample Input 1

```
-1 1
```

Sample Output 1

```
0
```

Sample Input 2

```
-999 1000
```

Sample Output 2

```
1000
```

Almost blank page

Problem B
Pr

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

The Prüfer code of a labeled tree is a unique sequence associate with the tree. Assume that T is a labeled tree of n vertices, and the vertices of T are labeled $1, 2, \ldots n$. The Prüfer code for T has length $n - 2$ and can be generated by the following procedure:

1. Initialize `arr` as an empty list.
2. Let x be a vertex which has exactly one neighbor. If there are multiple candidates of x, then pick the one with minimum index.
3. Append the index of x's neighbor to `arr`.
4. Remove x from T.
5. If T still has more than two vertices, go to step 2.

When the procedure terminated, the sequence stored in `arr` is the Prüfer code of T. Note that every Prüfer code can be recover to the unique tree. Therefore, we can represent an n-vertex tree with a Prüfer code of length $n - 2$.

Now we use the function `randPruder` in the following Python code to produce a random Prüfer code of an `n`-vertex tree where `randint(L,R)` is a function returning a uniformly random number between `L` and `R`, inclusively.

```
from random import randint

fun randPrufer(n):
    arr = []
    for i in range(n-2):
        arr.append(randint(1,n))
    return arr
```

Let T_R be the labeled tree corresponding to the random Prüfer code produced by `randPrufer(`n`)`. In this problem, a valid vertex coloring of T_R with c colors $C_1, C_2, \ldots, C_c$ is defined as follows.

- Every vertex in T_R must be colored in one of $C_1, C_2, \ldots, C_c$.
- Two endpoints of any edge in T_R are not colored in the same color.

Please write a program to compute the expected number of ways of valid vertex coloring of T_R.

Input Format

The input has exactly one line. That line contains exactly two integers n and c separated by a blank.

Output Format

If the answer is not an integer, then output "`fractional`" on a line. Otherwise, output the answer modulo $10^9 + 7$, since the answer might be very large.

Technical Specification

- $2 \leq n \leq 8$
- $1 \leq c \leq 10^9$

Sample Input 1

```
1 1
```

Sample Output 1

```
1
```

Sample Input 2

```
2 1
```

Sample Output 2

```
0
```

Sample Input 3

```
2 2
```

Sample Output 3

```
2
```

Problem C
Cakes

Time limit: 5 seconds
Memory limit: 256 megabytes

Problem Description

There are n different types of cakes, and the weight of a piece of the i-th type of cakes is w_i grams. Assume we have an unlimited supply of all kinds of cakes.

q monsters are hungry, and Cindy will feed them with cakes. The monsters never waste, so they will eat entire piece of cake. The monsters are curious, so they prefer to eat different types of cakes. Every monster eat any type of cakes at most one piece. That is, no monster eat two or more pieces of any one type of cakes. Each monster has its own preference on cakes. The j-th monster hates the x_j-th type of cakes, so the j-th monster never eats the x_j-th type of cakes. With the above contraints, it is not hard to observe that every monster will eat a subset of the n types of cakes. Moreover, every monster eats exactly one piece of each type in the subset.

To keep the monster happy, Cindy must feed the j-th monster with exactly W_j grams of cakes. If the j-th monster does not have exactly W_j grams of cake or Cindy feeds the j-th monster with the x_j-th type of cakes, then the j-th monster will become angry. Cindy wonders how many different sets of cakes she can feed the j-th monster without make it angry.

Please write a program to help Cindy.

Input Format

The first line contains one integer n indicating the number of types of cakes. The second line contains n integers $w_1, w_2, \ldots, w_n$ separated by blanks. The weight of a piece of the i-th types of cakes is w_i grams for $i \in \{1, 2, \ldots, n\}$. The third line contains one integer q indicating the number of monsters. The remaining part of the input contains q lines. The j-th of those lines contains two integers x_j and W_j separated by a blank. The j-th monster hates the x_j-th type of cakes and becomes angry if Cindy does not feed it with exactly W_j grams of cakes.

Output Format

For each monster, output the number of different ways to feed it without making it angry. If the number is greater than 998244353, output it modulo 998244353.

Technical Specification

- $1 \leq n \leq 10^5$
- $1 \leq \sum_{i=1}^{n} w_i \leq 10^5$
- $0 < w_i$ for $1 \leq i \leq n$.
- $1 \leq q \leq 10^6$
- $1 \leq x_i \leq n$
- $1 \leq W_i \leq 10^9$

Sample Input 1

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

Sample Output 1

```
0
2
1
```

Problem D
Decompose

Time limit: 2 seconds
Memory limit: 256 megabytes

Problem Description

A multiset of two elements $\{x, y\}$ is a good pair if $x = y$. A multiset of three elements $\{x, y, z\}$ is a good triple if $x = y = z$ or $\{x, y, z\} = \{\min(x, y, z), \min(x, y, z) + 1, \min(x, y, z) + 2)\}$. In this problem, a partition of a multiset S is a set P of subsets of S such that $\bigcup_{p \in P} p = S$ and $\sum_{p \in P} |p| = |S|$. A partition P of a multiset S of $3n+2$ elements is a good if the following hold.

- P contains exactly one good pair.
- P contains exactly n good triples.

Given a positive integer n and a multiset S of $3n+2$ elements where every element in S repeats at most four times in S. Please write a program to determine whether one can decompose S into a good partition.

Input Format

The first line contains a positive integer n indicating the number of elements of the multiset S is $3n + 2$. The second lines contains $3n + 2$ integers $a_1, a_2, \ldots, a_{3n+2}$ separated by blanks. The multiset S is defined as $\{a_1, a_2, \ldots, a_{3n+2}\}$.

Output Format

If one can decompose S into a good partition, then output "`YES`" (without quotes). Otherwise, output "`NO`" (without quotes).

Technical Specification

- $1 \leq n \leq 50000$
- $1 \leq a_i \leq 10^9$ for $1 \leq i \leq 3n + 2$.
- There does not exist x such that $\{x, x, x, x, x\} \cap S = \{x, x, x, x, x\}$.

Sample Input 1

```
1
1 2 2 3 3
```

Sample Output 1

```
NO
```

Sample Input 2

```
1
1 2 2 2 3
```

Sample Output 2

```
YES
```

Almost blank page

Problem E
Eddy's Secret

Time limit: 15 seconds
Memory limit: 256 megabytes

Problem Description

There is a secret string s with only English letters in lowercase. And Eddy wonders how many different strings satisfying the following conditions.

- The string is of length x.
- The string consists of only English letters in lowercase.
- The secret string is a substring of the string.

Because the number can be very large, please output the number modulo 998244353.

Input Format

The first line contains the secret string s. The second line contains a positive integer x.

Output Format

Output the answer on a line.

Technical Specification

- s consists of only English letters in lowercase.
- $1 \leq |s| \leq 200$
- $1 \leq x \leq 10^9$

Sample Input 1

```
aaa
4
```

Sample Output 1

```
51
```

Almost blank page

Problem F
Apple

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Iðunn is the goddess of apples of youth in Norse mythology. It is said that the gods in Norse mytholody must eat apples of youth to stop aging. Loki once decoyed Iðunn out of Asgard (where the gods live). Without Iðunn's apples, the gods became old and grey. They demanded Loki to return Iðunn. Loki turned her into a nut and took her back to Asgard.

Iðunn's job is collecting apples of youth in the garden of Asgard. There are n apple trees (numbered from 1 to n) and $n-1$ trails (numbered from 1 to $n-1$) in the garden. Each trail ends at two different apple trees. For any pair of apple trees, Iðunn can move from one to another along the trails in the garden. Suppose that apple tree i has d_i trails ending at it. Iðunn collects apples according the following procedure everyday.

1. Iðunn randomly moves to a tree with probability $\frac{d_i}{2n-2}$.
2. Iðunn uniformly randomly picks a trail ending at her current position, then she moves to the other end of the trail.
3. Collect apples from the tree at her position. If the tree is numbered i, then she should collect a_i apples.
4. Repeat the previous two steps for k times.

Write a program to compute the excepted number of apples daily collected by Iðunn.

Input Format

The first line contains two numbers n and k separated by blanks. There are n apple trees. k is the parameter in Iðunn's daily collecting procedure. In the second line, there will be n numbers a_i indicating how many apples should be collected in the third step of the procedure. In the j-th of the following $n-1$ lines, there will be two numbers u_j and v_j ($1 \le u_j, v_j \le n$) indicating that the j-th trail ends at tree u_j and tree v_j.

Output Format

Output the expected number of apples collected by Iðunn. An absolute error of 5×10^{-3} is acceptable.

Technical Specification

- $2 \le n \le 10^5$
- $1 \le k \le 10^5$
- $1 \le a_i \le 100$ for $i \in \{1, 2, \ldots, n\}$.
- $u_j, v_j \in \{1, 2, \ldots, n\}$ for $j \in \{1, 2, \ldots, n-1\}$.

Sample Input 1

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

Sample Output 1

```
8.00
```

Problem G
Vintage Screen

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

The classic computer screens can only display text. Typically, such a screen has 24 rows and 80 columns. Vintage Display Technology Company (VDTC) plans to produce some old fashioned computer screen. But it is no cool to display only 24 rows on one screen. VDTC wants their screens to have enough rows for some classical literature under the constraint that each row may only display 80 characters. To ideally display an article on the screen, the following rules may not be violated.

- The words must displayed in the order in the article.
- Any of the words must be displayed on exactly one row.
- Any two consecutive words must be separated by a blank or a newline.

Your task is to help VDTC to compute how many rows are required to display an article. If you manage to write such a program, then VDTC may use the program to analyze articles and determine the ideal number of rows of their new products.

Input Format

The first line contains a positive integer n indicating the number of words in the article. Then n lines follows. The i-th line of the following n lines contains the i-th word in the article.

Output Format

Output the number of rows required to ideally display the article.

Technical Specification

- $n \leq 10^5$
- The words consist of only English letters in lowercase.
- The length of any word in the article is at most 80.
- The total length of the words in the article is at most 10^5.

Sample Input 1

```
4
helloworldhelloworld
helloworldhelloworld
helloworldhelloworld
helloworldhelloworld
```

Sample Output 1

```
2
```

Almost blank page

Problem H
Yet Another Vintage Screen

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

The classic computer screens can only display text. Typically, such a screen has 24 rows and 80 columns. Vintage Display Technology Company (VDTC) plans to produce yet another kind of old fashioned computer screen. But it is no cool to display only 80 columns on one screen. VDTC wants their screens to have enough columns for some classical literature under the constraint that each screen may only display 24 rows. To ideally display an article on the screen, the following rules may not be violated.

- The words must displayed in the order in the article.
- Any of the words must be displayed on exactly one row.
- Any two consecutive words must be separated by a blank or a newline.

Your task is to help VDTC to compute how many columns are required to display an article. If you manage to write such a program, then VDTC may use the program to analyze articles and determine the ideal number of columns of their new products.

Input Format

The first line contains a positive integer n indicating the number of words in the article. Then n lines follows. The i-th line of the following n lines contains the i-th word in the article.

Output Format

Output the number of columns required to ideally display the article.

Technical Specification

- $n \leq 10^5$
- The words consist of only English letters in lowercase.
- The total length of the words in the article is at most 10^5.

Sample Input 1

```
4
helloworldhelloworld
helloworldhelloworld
helloworldhelloworld
helloworldhelloworld
```

Sample Output 1

```
20
```

Almost blank page

Problem I
Bamboo Rats

Time limit: 5 seconds
Memory limit: 256 megabytes

Problem Description

There are n bamboo rats and n burrows (holes in the ground dug by animal) on the flat. We specify their positions with 2D coordinates. Bamboo rat i is at (x_i, y_i), and burrow j is at (p_j, q_j). A bamboo rat needs to consume d^2 grams of bamboo leaves to move d meters. The bamboo rats have to stay in the burrows to wait out the winter season. Otherwise, the Chinese will catch them and eat them. It is too cruel to watch the bamboo rat in the videos uploaded by mukbangers. So you decide to help these n bamboo rats to move to the burrows. However, every burrow may only accommodate one bamboo rat. You have to find a way to assign the burrows to the bamboo rats. Moreover, you want to save the bamboo leaves for overwintering. Please write a program to compute the least amount of bamboo leaves consumed by the bamboo rats.

Input Format

There are $2n + 1$ lines. The first line contains an integer n indicating the number of bamboo rats and also the number of burrows. For $i \in \{1, 2, \ldots, n\}$, the $(i + 1)$-th line contains two integers x_i and y_i separated by a blank. For $j \in \{1, 2, \ldots, n\}$, the $(j + n + 1)$-th line contains two integers p_i and q_i separated by a blank.

Output Format

Output the least amount of bamboo leaves consumed by the bamboo rats in grams.

Technical Specification

- $1 \le n \le 500$
- $x_i, y_i \in \{1, 2, \ldots, 10^6\}$ for $i \in \{1, 2, \ldots, n\}$.
- $p_j, q_j \in \{1, 2, \ldots, 10^6\}$ for $j \in \{1, 2, \ldots, n\}$.
- For $i \neq j$, we have $(p_i, q_i) \neq (p_j, q_j)$.

Sample Input 1

```
2
0 0
0 1
1 0
1 1
```

Sample Output 1

```
2
```

Almost blank page

Problem J
Joyful Path

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

You are planing your next family trip to a foreign country. The number of days of this trip will be between d days to D days. This foreign country has n cities (numbered from 0 to $n-1$) such that there are direct flights between your home town and them. The trip can start and end in any of these n cities.

Your family loves taking trains. During the trip, you may stay in a city without any joy or move from one city to another one via train. You notice that there are m train routes via which your family will have some joy.

You numbered these m route from 1 to m. Route i starts from city u_i to city v_i. It gives your family j_i joyness and takes d_i days to travel. Please find the maximum average joyness (the total joyness divided by the number of days of your trip) travel plan.

Input Format

The first line of the input contains four positive integers n, m, d, D. n is the number of cities. m is the number of joyful routes. The length of your family trip can only be d to D days. Then m lines follows. The i-th of them contains four integers u_i, v_i, j_i and d_i described in the problem statement.

Output Format

Output the total joyness J and the length of your family trip L separated by a blank. This must maximize the average joyness. If there are multiple solution, output the one with L minimized.

Technical Specification

- $1 \leq n \leq 100$
- $1 \leq m \leq 1000$
- $1 \leq d \leq D \leq 100$
- $0 \leq u_i < n$ for $1 \leq i \leq m$
- $0 \leq v_i < n$ for $1 \leq i \leq m$
- $1 \leq j_i \leq 1000$ for $1 \leq i \leq m$
- $1 \leq d_i \leq 1000$ for $1 \leq i \leq m$

Sample Input 1

```
4 5 10 10
1 2 3 4
2 3 1 2
3 0 1 3
0 1 4 2
2 0 1 9
```

Sample Output 1

```
8 10
```

Problem K
Ideal Triangle Checker

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

We say a point (x, y) on the 2D-plane is a lattice point if x and y are integers. We say a non-degenerated triangle is a ideal triangle if all of its vertices are lattice points. You are given an ideal triangle n. Please compute the number of lattice points on its boundary.

Note: the area of a non-degenerated triangle is positive.

Input Format

The input consists of three lines which are distinct. Each of them contains two integers x and y indicating (x, y) is the coordinate of some vertex of the ideal triangle.

Output Format

Output the number of lattice points lying on the boundary of the given ideal triangle.

Technical Specification

- $0 \leq x \leq 10^9$
- $0 \leq y \leq 10^9$

Sample Input 1

```
0 0
2 2
2 0
```

Sample Output 1

```
6
```

Almost blank page

Problem L
Runs

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

A string is a sequence of characters. A binary string is a string consisting of only 0's and 1's. Suppose you are given a binary string $s = c_1c_2\cdots c_n$ of length n. A run in s is a continuous segement $c_bc_{b+1}\cdots c_{e-1}c_e$ such that the following conditions hold.

- $b = 1$ or $c_{b-1} \neq c_b$.
- $e = n$ or $c_{e+1} \neq c_e$
- $c_b = c_{b+1} = \cdots = c_{e-1} = c_e$

For example, `11011100101` has 7 runs.

For every non-negative integer x, we can always represent x by a binary string using the binary numeral system. Let $t(x)$ be the shortest binary string representing x in the binary numberal system. Write a program to compute the number of runs of the string $t(B)t(B+1)\cdots t(E)$.

Input Format

The input contains two integers B and E separated by one blanks.

Output Format

Output the number of runs of $t(B)t(B+1)\cdots t(E-1)t(E)$.

Technical Specification

$0 \leq B \leq E \leq 10^9$.

Sample Input 1

```
1 1000000000
```

Sample Output 1

```
14570502158
```

Sample Input 2

```
1 5
```

Sample Output 2

```
7
```

Almost blank page

Problem M
Monotone Chain

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Monotone chain algorithm is a popular method to construct a 2D convex hull. In this problem, an upper monotone chain of length n is a sequence of points $(x_1, y_1), \ldots, (x_n, y_n)$ on the 2D-plain such that

- $x_1 < x_2 < \cdots < x_n$.
- $\frac{y_{i+1}-y_i}{x_{i+1}-x_i} < \frac{y_i-y_{i-1}}{x_i-x_{i-1}}$.

Please write a program to generate an upper monotone chain of length n.

Input Format

The input is one line containing an positive integer n.

Output Format

Output n lines. The i-th of them consists of two integer x and y separated by a blank where (x, y) is the coordinate of the i-th point of an upper monotone chain.

Technical Specification

- $1 \leq n \leq 10^6$
- You may only output tokens no longer than 15.

Sample Input 1

```
3
```

Sample Output 1

```
0 0
1 1
2 0
```