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

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

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: 10 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

```
8 10 3
4 8 8 3 3 5
4 2 2 1 1 7 7 5
2 6 6 7
1 3
4 5
1 6
8 8
```

Sample Output 1

```
4
5
0
```

Almost blank page

Problem E
P-adic Equation

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

Kurt's got some equations to solve, help him out! You will be given the following items:

- A prime p, integer $2 \le n \le 200$, integer $1 \le k$
- A square matrix $A \mod p$ of size $n \times n$
- An integer vector $0 \le b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$ of size $n \times 1$
- An integer k

Matrix A is guaranteed to be invertible mod p. Try to solve for $x \mod p^k$ where

$$Ax \equiv b \mod p^k$$

It is guaranteed that $p^k < 2^{31}$ and $\forall b_i : 0 \le b_i < 2^{31}$

Input Format

The first line contains p, n, k seperated by single blanks. The second line contains $b_1, \ldots, b_n$ the elements of b. Then for the next n lines, the i-th line contains $a_{i,1}, a_{i,2}, \ldots, a_{i,n}$, where $a_{i,j}$ is the i-th row j-th column element of A.

Output Format

Output a line containing $x_1, \ldots x_n$ seperated by blanks.

Sample Input 1

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

Sample Output 1

```
0 2
```

Sample Input 2

```
3 2 1
0 2
1 1
1 2
```

Sample Output 2

```
1 2
```

Sample Input 3

Sample Output 3

```
3 2 2
0 2
1 1
1 2
```

```
7 2
```

Sample Input 4

```
3 2 3
0 2
1 1
1 2
```

Sample Output 4

```
25 2
```

Problem F
Forming Number

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

You are given a positive integer n. You have infinite supplies of 1's, additions, multiplications and brackets for composing an expression. What is the least total number of additions and multiplications required to compose an expression evaluated into n?

For example, $n = 9 = (1 + 1 + 1) \times (1 + 1 + 1)$ can be evaluated with 5 operations and it is impossible to use 4 operations to compose an expression evaluated into 9. Therefore, you should output 5. Note that concatenating 1's is not allowed.

Input Format

First line contains the number of testcases T. Each testcase is a line containing a positive integer n.

Output Format

For each testcase, output a number indicating the minimum total number of additions and multiplications required.

Technical Specification

- $1 \le T \le 10^4$
- $1 \le n \le 10^5$

Sample Input 1

```
10
1
2
5
11
22
55
100
222
555
1000
```

Sample Output 1

```
0
1
4
7
9
11
13
15
18
20
```

Almost blank page

Problem G
Geometric Center

Time limit: 1 second

Memory limit: 256 megabytes

Problem Description

We say a point (x^*, y^*) is a geometric median of n points $(x_1, y_1), \ldots, (x_n, y_n)$ on the 2D-plane if $\sum_{i=1}^{n} \sqrt{(x^* - x_i)^2 + (y^* - y_i)^2} = \min_{x,y} \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2}$.

It is probably too hard for you to find a geometric median in general. So please try to solve the cases for $n = 4$.

Input Format

The input consists of four lines. The i-th line contains two positive integer x_i and y_i indicating the i-th point is (x_i, y_i).

Output Format

Output x' and y' separated by a blank where (x', y') should be a geometric median.

Technical Specification

- $n = 4$
- $0 \leq x_i \leq 1000$ for $1 \leq i \leq n$
- $0 \leq y_i \leq 1000$ for $1 \leq i \leq n$
- The judge script will consider it is correct if your output x' and y' satisfies:

$$\sum_{i=1}^{n} \sqrt{(x' - x_i)^2 + (y' - y_i)^2} \leq 1.000001 \cdot \min_{x,y} \sum_{i=1}^{n} \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

Sample Input 1

```
0 0
1 0
0 1
1 1
```

Sample Output 1

```
0.5 0.5
```

Almost blank page

Problem H
Construct DAG

Time limit: 1 second
Memory limit: 256 megabytes

Problem Description

This problem is about to construct a simple directed acyclic graph (DAG) of n vertices and m edges. The graph must have the vertex set be $\{1, \ldots, n\}$. Any edge (u, v) in this graph must satisfy $u < v$. Any vertex u may have at most k out-going edges. And multiple edges are not allowed.

You are given three numbers n, k, a where n is the number of vertices and k is the limit of out-going edges. Please determine whether there is a simple DAG such that there are at least a^n paths from 1 to n.

Input Format

The input is one line containing three numbers n, k, a.

Output Format

If there is no such DAG, output `-1`. Otherwise, output $n - 1$ lines. The i-th line is a list of vertices that is adjacent to i. Separate the vertices with a blank.

Technical Specification

n and k are no more than 1000, and a is a positive number in $[1, 10^9]$. You may assume that double precision floating point numbers are accurate enough.

Sample Input 1

```
4 3 2.34567
```

Sample Output 1

```
-1
```

Sample Input 2

```
4 3 1
```

Sample Output 2

```
2
3
4
```

Problem I
Ideal Triangle

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 an ideal triangle if all of its vertices are lattice points. You are given a positive n and asked to generate an ideal triangle such that there are exactly n lattice points on its boundary.

Note: the area of a non-generated triangle is positivel.

Input Format

The input is a line containing a positive integer n.

Output Format

Output three lines which are distinct. Each of them contains two integers x and y indicating (x, y) is the coordinate of an ideal triangle. This ideal triangle must have n lattice points on its boundary.

Technical Specification

- $3 \le n \le 10^{32}$
- $0 \le x \le 10^{64}$
- $0 \le y \le 10^{64}$

Sample Input 1

```
6
```

Sample Output 1

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

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

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

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

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

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	Sample Output 1
3	0 0 1 1 2 0