# Technology Arena - Day 1

## A. Debug

3 seconds, 256 megabytes

Code debugging is a common thing programmers do. Some do it better, some do it worse, hence we'll denote each programmer's debugging skill as $d$. When a programmer whose debug skill is $d$ comes upon a problem of difficulty $h$, it'll take him $\texttt{floor}(\frac{h}{d})$ time to finish debugging the code.

It's also common knowledge that plastic ducks help with code debugging. Throughout the debugging process, programmers like to explain their thought process to their ducks, and in doing so, the ducks inadvertently improve the programmer's skill. The bigger the duck, the more it helps the programmer. If the volume of a duck is denoted as $v$, then the programmer's skill becomes $d + v$.

There are currently $n$ debugging problems in need of a solution and $n$ programmers at the office in Rovinj, Croatia. Each programmer of debug skill $d_i$ was assigned a debugging problem of difficulty $h_i$. There are also $n$ ducks at the office, each with a volume of $v_i$, that can be divided amongst the programmers.

Each programmer can only debug the problem it was assigned to him specifically, meaning that no reassignments of problems can be made. Programmers are not allowed to help each other with debugging. Once the ducks have been divided amongst the programmers, any duck-swapping is strictly forbidden.

The programmers have a mutual agreement that they'll go for a swim once everyone is finished with their respective debugging problem. What is the earliest possible time the programmers can go for a swim?

### Input

The first line contains integer $n$ ($1 \leq n \leq 10^5$), the number of programmers.

The second line contains $n$ integers, the debug skill $d_i$ ($1 \leq d_i \leq 10^6$) of $i$-th programmer.

The third line contains $n$ integers, the difficulty $h_i$ ($1 \leq h_i \leq 10^9$) of the problem which was assigned to the $i$-th programmer

The fourth row contains $n$ integers, the volumes of ducks $v_i$ ( $1 \leq v_i \leq 10^6$).

### Output

One integer, the earliest possible time when all programmers will have finished their debugging problems.

### Scoring

(21 points): $1 \leq n \leq 8$.

(35 points): $1 \leq n \leq 500$.

(44 points): $1 \leq n \leq 10^5$.

**input**
```
4
10 100 25 20
301 1502 3 329
50 10 4 20
```
**output**
```
10
```

**input**
```
2
1 29
40 329
1 3
```
**output**
```
10
```

**Explanation of the first sample**

The first person should get the duck with volume $20$. Their total debug skill is $10 + 20 = 30$, so they finish in moment $\texttt{floor}(301/30) = 10$.

The second person should get the duck with volume $50$. They finish in moment $10$.

The third person should get the duck with volume $4$. They finish in moment $0$.

The fourth person should get the duck with volume $10$. They finish in moment $10$. Note that we rounded that down (and not to the closest integer).

The first moment at which everyone is finished is $10$.

**Explanation of the second sample**

The first programmer should get the second duck, while the second programmer should get the first duck. Both will finish at moment $10$.

## B. Turtle

1 second, 256 megabytes

Bob has made a playground for his pet turtle in his backyard. The backyard can be viewed as an infinitely large matrix on which Bob had placed leaves of spinach in such a way that each column has at most one leaf of spinach. The turtle has to collect the leaves of spinach, its starting point is the left-most leaf, while its ending point is the right-most leaf in the matrix. In between her starting point and her ending point, the turtle is allowed to move in three directions: up, down, and right. Each of the turtle's moves lasts exactly one second. The act of collecting a leaf is instant and it lasts 0 seconds. Since the turtle might starve to death if it doesn't eat for a while, Bob put leafs such that the distance $d$ between the neighbouring leaves is at most a 3-day long turtle's trip.

What is the fastest time in which Bob's turtle can collect all of the spinach leaves? How many different paths can the turtle take to collect all of the leaves, while collecting them in the fastest possible time? Print the result as modulo $10^9 + 7$.

### Input

The first line contains, $n$, $2 \leq n \leq 3 \cdot 10^5$, the number of leaves Bob has dispersed in his backyard.

The next $n$ lines contain two numbers: $0 < x_i < 10^9$ and $0 < y_i < 10^9$, the coordinates of the $i$-th leaf. The coordinates of leaves are given in ascending order with regards to the $x$ coordinate.

It is guaranteed that the distance $d_i$ between neighbouring leafs is at most 3 days. Formally, for each valid $i$, it is guaranteed that $|x_i - x_{i+1}| + |y_i - y_{i+1}| \leq 259200$.

### Output

The first line contains the minimum number of seconds required to collect all of the leaves, modulo $10^9 + 7$.

The second line contains the number of different paths the turtle can take, modulo $10^9 + 7$.

### Scoring

(31 points): $n \leq 50$, $d \leq 50$.

(24 points): $n \leq 3 \cdot 10^5$, $d \leq 1000$.

(45 points): $n \leq 3 \cdot 10^5$, $d \leq 259200$.

**input**
```
4
2 5
3 3
5 3
6 4
```
**output**
```
7
6
```

**input**
```
5
13047 9550
13055 9574
13095 9580
13127 9595
13173 9592
```
**output**
```
174
443642658
```

The turtle starts from (2,5) and has to get to position (6,4) while collecting extra leaves in (3,3) and (5,3). A total of 7 seconds is needed to get all the leaves.

One optimal path which requires 7 seconds:
$(2,5) \to (3,5) \to (3,4) \to (3,3) \to (4,3) \to (5,3) \to (5,4) \to (6,4)$
.

There are 6 different paths which are 7 seconds long. They are:

- (2,5) → (3,5) → (3,4) → (3,3) → (4,3) → (5,3) → (5,4) → (6,4)
- (2,5) → (2,4) → (3,4) → (3,3) → (4,3) → (5,3) → (5,4) → (6,4)
- (2,5) → (2,4) → (2,3) → (3,3) → (4,3) → (5,3) → (5,4) → (6,4)
- (2,5) → (3,5) → (3,4) → (3,3) → (4,3) → (5,3) → (6,3) → (6,4)
- (2,5) → (2,4) → (3,4) → (3,3) → (4,3) → (5,3) → (6,3) → (6,4)
- (2,5) → (2,4) → (2,3) → (3,3) → (4,3) → (5,3) → (6,3) → (6,4)

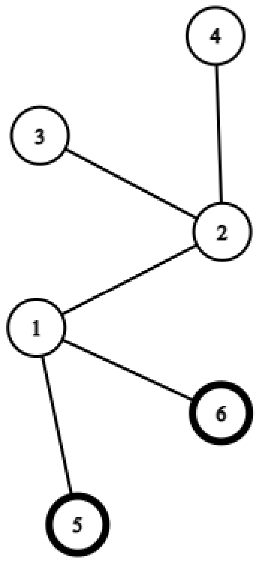
## C. Promenade

4 seconds, 256 megabytes

A promenade consists of $n$ interconnected cottages. There is exactly one path connecting each pair of cottages. Over time, the promenade became too big to maintain and now the promenade's Board wants to cut down maintenance costs. However, they still want to provide enough interconnected cottages so that the average visitor's experience is not diminished. They came up with the following idea: in order to succeed in their goal, they want to shut down the minimum number of cottages so that the distance between any two remaining cottages is not greater than the number of cottages an average visitor visits.

The Board team looked into their cottage-visitation records and found how many cottages an average visitor visits, and then they denoted it as $m$.

Given a map of the promenade and the number of cottages visited by an average visitor $m$, find the minimum number of cottages that have to be closed.




Promenade from the first example; in this case, cottages 5 and 6 can be closed to achieve the optimal result.

### Input

The first line contains $2 \le n \le 1000$, the number of cottages, followed by $m$, $1 \le m \le n$, the number of cottages visited by an average visitor.

The next $n-1$ lines contain two numbers: $u_i$ and $v_i$, $1 \le u_i, v_i \le n$, the road between cottages $u_i$ and $v_i$.

### Output

A single number $k$ which indicates how many cottages have to be closed so that the Board's goal is achieved.

**input**
```
6 2
1 2
2 4
2 3
1 5
6 1
```
**output**
```
2
```

**input**
```
2 1
1 2
```
**output**
```
0
```

## D. spam

1 second, 256 megabytes

Alice wrote a brilliant essay on the topic of saving the trees and other nature's blessings. Bob didn't think so though, so he decided to fill her essay with spam - literally. Bob took Alice's essay and added the letters `'s'`, `'p'`, `'a'` and `'m'` at random locations throughout her essay, and he did that zero or more times.

For example, if Alice's essay was `prometheusistheoldesttreeintheworld` Bob could add any of the aforementioned letters (zero or more times) so the modified essay could be as follows:
prometheus**p**istheol**p**dest**aaaaa**treein**aa**the**mmmm**world

Devastated, Alice decided to fix her essay but she needs help. The first thing she wants to find out value $k$ which denotes how many spam words could be constructed from her essay? Each spam word must have one `'s'`, one `'p'`, one `'a'`, and one `'m'`, which appeared in the modified essay in that order. I.e. `'s'` had to appear before `'p'`, which had to be before `'a'`, which had to be before `'m'`. Also, no letter can be used in two different spam words. For example, modified essay `ssppaamm` has $k$ value equal to $2$, while modified essay `amspamspm` has $k$ equal to $1$. Modified essay `prometheu`**`sp`**`i`**`s`**`theol`**`p`**`dest`**`aa`**`aaatreeinaathe`**`m`**`mm`**`m`**`world` has $k$ equal to $2$, with highlighted one possibility for spam words.

### Input

The first line contains integer $n$ ($1 \le n \le 3 \cdot 10^5$), the length of Alice's essay.

The second line contains $n$ characters (lowercase ASCII characters) which represent Alice's essay after Bob added the spam.

### Output

First Integer $k$, maximal number of spam words which could be created from the modified essay.

### Scoring

(21 points): $1 \le n \le 300$.

(79 points): $1 \le n \le 3 \cdot 10^5$.

**input**
```
8
ssppamam
```
**output**
```
2
```

**input**
```
15
spspaxaaspmmmmm
```
**output**
```
2
```

**input**
```
9
amspamspm
```
**output**
```
1
```

## E. The Opposites Attract

1 second, 256 megabytes

We are preparing a yes/no personality test for couples. The test relies on the *opposites attract* rule. When a couple answers a question differently, the couple gets $1$ point, and if the answers match, the couple gets no points.

There are $n$ couples taking part in the test, each couple getting asked a total of $q$ yes/no questions. As there are going to be numerous couples, we would like to have a prepared grading mechanism.

### Input

The first line contains $n$, the number of couples ($1 \le n \le 1000$).

The second line contains $q$, the number of questions asked in the test ( $1 \le q \le 1000$).

$2n$ lines follow, where line $l_i$ represents the answers from the first partner in the $i$-th couple, and line $l_{i+1}$ represents the answers collected from the other partner of the $i$-th couple. Each line contains $q$ characters, where `0` denotes a `NO` answer and `1` denotes a `YES` answer.

### Output

For each of the $n$ couples print $p_i$, where $p_i$ is the number of points the $i$-th couple got.

**input**
```
2
5
11001
10011
00000
11111
```
**output**
```
2
5
```

In this example, there were two couples getting asked $5$ questions.

The first couple answered the second and fourth question differently, so their score is $2$.

The second couple answered all the questions differently, hence their score is $5$.

## F. Gary, The Robot Vacuum Cleaner

1 second, 256 megabytes

Gary is a state-of-the-art robot vacuum cleaner whose job is to clean rectangle-shaped rooms. He is fully capable of learning what's in his surroundings and, once he has learned the entirety of his surroundings' floor plan, he always keeps a tab of how many times he has cleaned each spot. It is assumed that Gary's surroundings never change.

Once Gary had learned what's in his surroundings, you ordered him to clean $t$ times. Each order was sent as a sequence of four numbers which describe the rectangle's bounding box: the upper left and the bottom right point, respectively.

The floor plan can consist of obstacles. If a section of the floor plan is completely surrounded by obstacles, Gary will teleport to that section and clean it. He, however, cannot clean spots where any obstacles lay. Given the floor plan of size $n \times m$, where `#` denotes an obstacle and `.` denotes free room, print the map Gary will have stored in memory after $t$ cleaning tasks.

### Input

The first line contains $n$ and $m$, the number of rows and columns used to represent the floor plan. $1 \le n, m \le 50$.

The next $n$ lines contain $m$ characters (either `#` or `.`) describing the floor plan.

The next line contains $t$, $1 \le t < 10$, the number of cleaning tasks.

The next $t$ lines contain the bounding box of the rectangle where Gary has to clean, given as four numbers: $x_i$, $y_i$, $x_j$ and $y_j$; $0 \le x_i \le x_j < n, 0 \le y_i \le y_j < m$.

### Output

Print Gary's $n \times m$ floor plan where each element describes how many times Gary has cleaned that spot. Print obstacles as `#`.

**input**
```
3 5
.....
#####
.....
5
0 0 2 2
2 0 2 4
0 1 2 3
1 0 1 0
0 0 0 0
```
**output**
```
22210
#####
23321
```

After the first task, where Gary had to clean from (0, 0) to (2, 2), Gary's map looked like this:

```
11100
#####
11100
```

After the second task Gary's map looked like this:

```
11100
#####
22211
```

After the third task, Gary's map looked like this:

```
12210
#####
23321
```

The fourth task did not influence Gary's map.

After the fifth and final task, Gary's map looks like this:

```
22210
#####
23321
```

## G. The Age of a Tree

1 second, 256 megabytes

Alice was taking a walk one day when she came upon a magic tree. The magic tree is celebrating its birthday today and has told Alice that it will reveal its name if Alice successfully converts its age from years to minutes. In order to do that, the tree would first reveal its age (in years) digit by digit.

Alice lives in Wonderland where there are no leap years and each year consists of exactly 365 days. Each day has 24 hours and each hour has 60 minutes.

Help Alice calculate the tree's age $a$ and print it mod $10^9 + 7$.

### Input

The first line contains $n$, $1 \leq n \leq 50$, the number of digits the tree has spoken.

The second line contains $n$ digits expressed as words. The words spoken could be (exclusively) any of the following:

- `zero`
- `one`
- `two`
- `three`
- `four`
- `five`
- `six`
- `seven`
- `eight`
- `nine`

All digits are separated by a single whitespace character and all characters are lowercase.

### Output

Print $a$, the age of the tree expressed in minutes, mod $10^9 + 7$.

**input**

```
5
one two three zero zero
```

**output**

```
464879958
```

**input**

```
4
zero zero zero one
```

**output**

```
525600
```

## H. Farm

5 seconds, 256 megabytes

As an outdoor enthusiast, Alice has just purchased a beautiful lot near a forest. Alice would like to build a rectangular log cabin on her lot and make it as big as possible. However, Alice's parcel is filled with trees and she wants to disturb nature's blessings as little as possible. In order to fulfil her own wishes, she needs help finding the biggest rectangular clearance (the biggest rectangular area where no trees grow) on her lot where the cabin can be built.

For an $n \times n$ map of Alice's lot, find the biggest area where Alice can build her cabin.

### Input

The first line contains a integer $n$ ($1 \leq n \leq 1000$).

The next $n$ lines contain $n$ characters where trees are denoted as "#" and clearances as "."

### Output

Output maximal possible surface of a rectangle where Alice can build her cabin.

### Scoring

(27 points): $n \leq 50$.

(73 points): $n \leq 1000$.

**input**

```
4
##.#
...#
....
.###
```

**output**

```
6
```

Alice can build a farm on rows $2$ and $3$, and from column $1$ until row $3$. Total area is $2 \times 3 = 6$.

## I. Decorative Ribbons

1 second, 256 megabytes

Alice has prepared $n$ gifts for her friends and is now in the process of wrapping them. She has $m$ ribbons of equal length at her disposal. Since she loves all her friends equally, Alice wants to divide the ribbons among the gifts in such a way that each gift gets the same length of ribbons. To achieve that, it's obvious to Alice that she might have to cut the ribbons.

Considering that these ribbons are decorative in nature, she doesn't want to ruin them by making too many cuts. Help Alice find $c$, the minimum number of cuts she has to make so that each gift gets an equal length of ribbons.

### Input

The first and only line contains the number of gifts $1 \leq n \leq 10^9$, and $m$, $1 \leq m \leq 10^9$, the number of ribbons.

### Output

Print $c$, the minimum number of cuts Alice has to make.

**input**

```
3 5
```

**output**

```
2
```

**input**

```
5 3
```

**output**

```
4
```

**input**

```
1 1
```

**output**

```
0
```

**input**

```
6 3
```

**output**

```
3
```

In the first example, Alice has 3 gifts and 5 ribbons at her disposal. This means each gift should get $1 + \frac{2}{3}$ length of ribbons. To achieve that, Alice can leave 3 ribbons intact and make 2 cuts on the remaining 2 ribbons (each ribbons gets cut once).

In the final example, Alice has to cut all three ribbons in half to equally divide them among 5 gifts.

## J. Migrations

1 second, 256 megabytes

There is a city called Gamescity which has $n$ inhabitants. Some of them are not pleased with the city's air quality, so they've decided to move to the outskirts, specifically to $m$ different (smaller) settlements which are located near Gamescity.

The number of people who move to each of the $m$ settlements (in the span of a year) is denoted as $G_i$ $(i = 0...m)$. What are the odds $P$ (in percentages) that an inhabitant of Gamescity will move out of Gamescity and into the outskirts in the period of $x$ years if a person is currently living the Gamescity?

### Input

The first line contains $n$, the total population size of Gamescity $100 \leq n < 10^9$.

The second line contains $m$, the number of surrounding settlements $1 \leq n < 300000$.

The third line contains $x$, the number of years for which a percentage needs to be found $1 \leq x \leq 10$.

The fourth line contains a number of citizen $G_i$ ( $0 \leq G_i \leq n; i = 0...m$) that migrate to a village (separated by whitespaces).

### Output

Print the result in percentages $1 \leq P \leq 100$ with no decimals.

Take the floor of the value of the percentage e.g. $23.86\%$ results in an output that is equal to 23.

**input**

```
10000
2
5
100 1200
```

**output**

```
50
```

Gamescity has $10000$ inhabitants at the beginning. After $5$ years expected percentage that a person will leave Gamescity is $50.16\%$ which floored gives a result of $50$.