TAMU Spring 2019 Team Programming Contest

A. Adventurous Welcoming

1 second, 256 megabytes

Greetings, space cadet! You've been selected for a top-secret space mission. Over the course of this mission, you'll be using your programming skills to optimize spaceships, communicate with aliens, and defend the galaxy! For now, let's just introduce ourselves to the new planets we'll be visiting.

There are n solar systems on our list, each of which has some number of planets, p. We plan to visit every planet in every solar system on the list. Let's say "Hello" to these new worlds!

Input

The first line of input contains one integer, n $(1 \leq n \leq 10)$, giving the number of solar systems. The second line of input contains n integers, p ($1 \leq p \leq 10)$, giving the number of planets in each solar system.

Output

Output the line "Hello x worlds", where x refers to the total number of planets in all of the solar systems in the list.

input

```
4
3 1 8 3
```

output

```
Hello 15 worlds
```

input

```
2
10 2
```

output

```
Hello 12 worlds
```

input

```
1
1
```

output

```
Hello 1 worlds
```

In the first sample input, there are 4 solar systems, which contain 3, 1, 8, and 3 planets. The total number of planets equals 15 (3+1+8+3), so the line "Hello 15 worlds" is printed to the screen. In the second sample input, there are 2 solar systems with 10 and 2 planets, for a total of 12. In the third sample input, there is 1 planet.

B. Building Spaceships

1 second, 256 megabytes

You've arrived at your first solar system! Its planets are completely uninhabited but have lots of useful spaceship parts on them. Perhaps there was a recent battle. This makes for a great opportunity to build a fleet of spaceships from the parts you can salvage! The captain of your exploration ship has personally asked for your help in assembling a space fleet.

A spaceship is constructed from three parts: an engine, a thruster, and a chassis. A planet can produce one spaceship for every combination of engine, thruster, and chassis on the planet. For example, a planet with 3 engines, 5 thrusters, and 2 chassis would be able to produce 2 spaceships. The leftover parts, 1 engine and 3 thrusters, would not be enough to produce an additional spaceship because there would be no chassis.

Some of the planets have teleporters, which allow for parts to be exchanged between planets. Parts may only be exchanged between two planets if they both have a teleporter. Planets without teleporters are isolated and cannot send or receive parts at all. The teleporters may be used an unlimited number of times.

Determine the maximum number of spaceships which can be produced through optimal use of teleporters.

Input

The first line of input contains two integers, n and t $(1 \leq t \leq n \leq 100)$, providing the number of planets and number of teleporters. The second line of input contains t nonrepeating integers p $(1 \leq p \leq n)$, referring to the planets with teleporters on them. The following n lines of input each contain 3 integers, a, b, and c $(1 \leq a, b, c \leq 100)$, referring to the number of engines, thrusters, and chassis on each planet.

Output

Print a single integer giving the number of spaceships which can be produced.

input

```
4 2
1 3
4 0 2
1 2 3
2 6 3
3 2 4
```

output

```
8
```

input

```
5 4
1 2 3 5
5 4 4
1 1 3
3 2 6
3 2 4
1 2 3
```

output

```
11
```

In the first input, the optimal strategy is to teleport 1 engine from Planet 1 to Planet 3 and 2 thrusters from Planet 3 to Planet 1. Planet 1 will then have 3 engines, 2 thrusters, and 2 chassis, which allows it to produce 2 spaceships. Planet 2, which did not transfer any parts, will produce 1 spaceship. Planet 3 will have 3 engines, 4 thrusters, and 3 chassis, which allows it to produce 3 spaceships. Planet 4, which did not transfer any parts, will produce 2 spaceships. The total number of spaceships produced will be 8 (2 + 1 + 3 + 2).

C. Combustive Reaction

1 second, 256 megabytes

Excellent work on the space fleet! You've been promoted to first mate of your exploration ship. The captain has asked for your help once again, this time in mobilizing your newly created fleet.

A spaceship can be powered by five different fuel sources, each with their own one-letter abbreviation: Hydrogen (H), Oxygen (O), Carbon (C), Nitrogen (N), and Xenon (X). Every second, a spaceship burns through one pound of one type of fuel. Certain combinations of gases, if consumed in the right order, generate bonus energy. The space chemist has provided you with a table of these fuel combinations, along with their resulting bonus energy. Fuel combinations are allowed to overlap. For example, if the combination "HCH" generates a bonus energy of 4, the fuel sequence "HCHCH" would generate a total bonus energy of 8, since the "HCH" sequence appears twice.

Given a fuel sequence, determine the total amount of bonus energy generated, accounting for all occurrences of all fuel combinations.

Input

The first line of input consists of two integers, n ($1 \leq n \leq 100$) and k ($1 \leq k \leq 10$), representing the length of the fuel sequence and the number of fuel combinations. The next line consists of a string s of size n, representing the fuel sequence. The following k lines each consist of a fuel combination, which is a string a of size z ($2 \leq z \leq 10$) and an integer b ($1 \leq b \leq 10$), representing the fuel sequence and its bonus energy. All input strings will be unique.

Output

Output a single integer giving the total amount of bonus energy generated.

input

```
18 5
OHNCCCCCNXNXNOHXHX
XN 6
XNO 3
CC 5
OHX 4
CNXH 10
```

output

```
39
```

input

```
3 2
HHH
HH 2
NX 1
```

output

```
4
```

In the first sample input, the sequence "XN" occurs 2 times for a combined bonus of 2*6=12, the sequence "XNO" occurs 1 time for a combined bonus of 1*3=3, the sequence "CC" occurs 4 times for a combined bonus of 4*5=20, the sequence "CNXH" never occurs, and the sequence "OHX" occurs 1 time for a combined bonus of 1*4=4. The total bonus energy generated is 12+3+20+4=39.

D. Dark Network

1 second, 256 megabytes

The captain was impressed with your engine calculations, and word of your skills has reached the space commander himself! You've been granted full autonomy over your own space legion, with the freedom to work on any project you want. You've set out to develop a means of communicating with the other aliens in the galaxy.

The aliens communicate by sending out signals of dark matter at a certain wavelength. A brief amount of research reveals that the signals seem to be made out of abstract entities called components. The components of a signal are the maximal set of unique prime numbers such that its wavelength is divisible by each of its components. For example, a signal with a wavelength of 20 would have the components $\{2, 5\}$. A signal with a wavelength of 27 would only have the component $\{3\}$.

You plan to send out your own signals to the other aliens, and your scientists have proposed a wavelength for the signals. The quality of the communication between two signals based on the proportion of overlapping components, which is calculated as the number of overlapping components divided by the number of unique components between the two signals. Given the wavelengths of the alien signal and your own signal, determine the proportion of overlapping components.

Input

Input consists of two integers, a and b ($2 \leq a, b \leq 10^9$), representing the wavelenghs of the alien signal and your own signal.

Output

Output a floating point number giving the proportion of overlapping components between the two signals. Your answer will be considered correct if its absolute or relative accuracy is within 0.0001 of the actual answer.

input

```
60 26
```

output

```
0.25
```

input

```
36 58
```

output

```
0.3333333333333333
```

input

```
45 2
```

output

```
0.0
```

In the first sample input, the components of 60 and 26 are $\{2, 3, 5\}$ and $\{2, 13\}$, respectively. The set of overlapping components, $\{2\}$, is of size 1. The set of total components, $\{2, 3, 5, 13\}$, is of size 4. The proportion of overlapping components is $\frac{1}{4}$ = 0.25.

E. Enigmatic Aliens

1 second, 256 megabytes

The communications network was a success! You've received your first transmission from an unknown alien source. The message is a bit difficult to understand, because it seems to be comprised of words from multiple different alien languages. Fortunately, the space linguist has provided you with a few space dictionaries of the most common alien languages.

Some of the words in the message may be words in more than one alien language. If a word shows up in multiple space dictionaries, it might have multiple different translations. Thus, the overall message could have multiple different meanings, where each meaning is a unique sequence of English translations. If one of the words does not show up in any of the space dictionaries, the entire message has zero possible meanings.

Given the message and the space dictionaries, determine the number of possible meanings the message could have. As the number of possible meanings can be very large, it should be output modulo 1000000007 ($10^9 + 7$).

Input

The first line of input consists of two integers, n and k ($1 \leq n, k \leq 100$), representing the number of words in the message and the number of space dictionaries. The following line contains n strings, representing the message. The remainder of input will consist of the space dictionaries. Input for each space dictionary will consist of a line containing an integer x ($1 \leq x \leq 100$), representing the number of words in the space dictionary, followed by x lines each containing the string for the alien word and the string for its English translation. Alien words within a space dictionary will be unique but are not necessarily unique across other space dictionaries. Input strings will contain no more than 10 characters.

Output

Output a single integer giving the number of possible meanings the message could have, modulo 1000000007 ($10^9 + 7$).

input

```
4 3
blar vor zesh klon
4
klon protection
xelg hello
vor come
zesh in
2
zesh in
klon rage
3
klon peace
vor attack
blar we
```

output

```
6
```

input

```
3 2
xark fen jorg
2
xark robots
jorg attacking
3
jorg invading
scorx help
walv us
```

output

```
0
```

In the first input, "blar" has one possible translation ("we"), "vor" has two possible translations ("come", "attack"), zesh has one possible translation ("in"), and "klon" has three possible translations ("protection", "rage", "peace"). Thus, there are a total of 6 possible meanings for the message: "we come in protection", "we come in rage", "we come in peace", "we attack in protection", "we attack in rage", and "we attack in peace". The solution is $6\%1000000007 = 6$. In the second input, there is no known translation for the second word of the message, "fen". Thus, the message has no possible meanings.

F. Faraday Cage

1 second, 256 megabytes

From decoding the cryptic alien message, you've found out that the entire galaxy is under attack! The Robot Empire threatens to unleash a massive wave of electromagnetic radiation, exterminating all forms of life in the galaxy. You have a way of blocking this attack with a Faraday Cage, but you're not yet sure if you have enough material to protect the whole galaxy.

The Faraday Cage is to be constructed out of space blocks. Each space block is a cube with a length of one unit. The Faraday Cage must be in the shape of a hollow cube with a thickness of 1 unit. The number of blocks required to construct the Faraday Cage depends on its size. For example, the smallest possible Faraday Cage would have a side length of 3 units and require a total of 26 blocks: 9 on the bottom layer for a 3x3 base, 8 on the middle layer for a 3x3 base with the single middle cube removed, and 9 on the top layer for a 3x3 ceiling. As another example, the Faraday Cage with a side length of 4 units would require a total of 56 blocks: 16 on the bottom for a 4x4 base, 12 on each of the next 2 layers for a 4x4 middle section with a 2x2 section removed, and 16 on the top for a 4x4 ceiling. As one further example, the 5-unit Faraday Cage would consist of 98 blocks.

Given the number of space blocks available for use in construction, determine the side length of the largest possible Faraday Cage.

Input

Input consists of a single integer, n ($26 \leq n \leq 5 * 10^{18}$), representing the number of available space blocks. Note that input may exceed the boundaries of 32-bit integers.

Output

Output a single integer giving the side length of the largest possible Faraday Cage.

input

```
50
```

output

```
3
```

input

```
98
```

output

```
5
```

input

```
215
```

output

```
6
```

In the first sample input, the largest possible Faraday Cage that can be constructed using 50 or fewer blocks is the 3-unit Faraday Cage, which uses 26 blocks. In the second sample input, the largest possible Faraday Cage that can be constructed using 98 or fewer blocks is the 5-unit Faraday Cage, which uses 98 blocks.

G. Galactic Laser

1 second, 256 megabytes

Great work protecting the galaxy! Unfortunately, the Robot Empire still lurks in the shadows, preparing their next attack. In order to help defend against this menacing threat, you've been granted membership to the Galactic Council. Their current mission is to create a laser powerful enough to obliterate the home planet of the Robot Empire.

The laser will fire from the Galactic Headquarters to the Robot Empire home planet. Thus, the exact direction of the laser is fixed. However, there is a certain range of positions from which the laser can be fired, such that the target planet will still be hit. The positioning of the laser is quite important, because the target planet is surrounded by a protective field of asteroids. Although the laser can pass through the asteroids, its power diminishes for every asteroid in its path.

The laser can be fired from any point in the range $[l, r]$. There are n asteroids, each of which span the points in the bounds $[a, b]$ (note that different asteroids may have different bounds). If the laser is fired from point x, it will collide with any asteroid where x is in the range of the asteroid's bounds. The Galactic Council would like to determine the minimum number of asteroids which will be hit if the laser is positioned optimally.

Input

The first line of input consists of three integers, n, l, and r, giving the number of asteroids, the left bound of the laser, and the right bound of the laser, respectively ($0 \leq n \leq 10^5, 0 \leq l \leq r \leq 10^9$). The following n lines of input each contain two integers a and b, referring to the leftmost and rightmost points of each asteroid ($1 \leq a \leq b \leq 10^9$).

Output

Output a single integer giving the minimum number of asteroids which will be hit if the laser is positioned optimally.

input

```
6 4 14
10 15
3 8
9 17
6 8
2 21
0 4
```

output

```
2
```

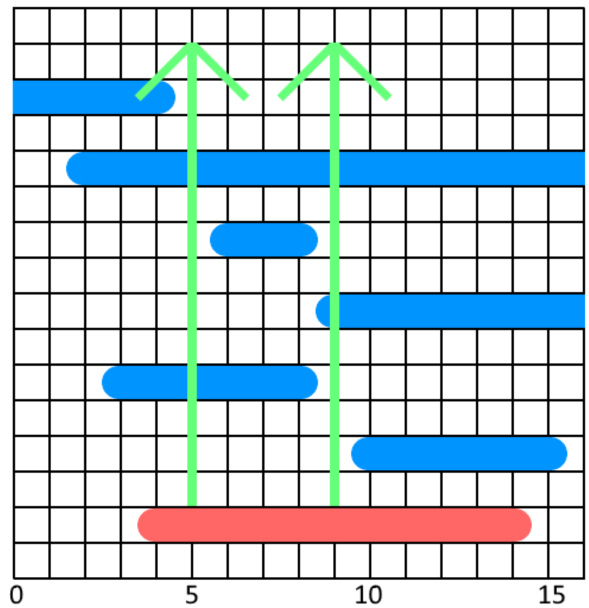
input

```
4 5 12
6 11
4 10
6 11
8 8
```

output

```
0
```

In the first sample input, the optimal strategy is to hit only 2 asteroids, by placing the laser at either position 5 (where it hits the second and fifth asteroids) or position 9 (where it hits the third and fifth asteroids). In the second sample input, the optimal strategy is hit 0 asteroids, by placing the laser at position 12. The image below depicts the first sample input, where the bounds of the laser are shown in red, the asteroids are shown in blue, and the optimal solutions are shown in green.



H. Harmonic Convergence

1 second, 256 megabytes

Despite your best attempts at optimizing the Galactic Laser, the Robot Empire survived the attack. They threaten once more to wipe out all life in the galaxy. Desperate for help, the Galactic Council has turned to a space psychic, who claims to know of a prophecy which will end the war. The space psychic speaks of a harmonic convergence, where all of the planets align in the home solar system of the Robot Empire. Your space astronomers have pointed out that the harmonic convergence would drastically reduce the amount of solar energy the Robot Empire can harvest. Perhaps there is some truth to the prophecy.

In the Robot Empire's home solar system, there are n planets. Each planet has a frequency f which refers to the number of years it takes for a complete rotation around the sun. The space psychic assures you that the harmonic convergence will occur at some point in the future. Assuming her prediction is correct, determine the amount of time in between each harmonic convergence. Note that, in between two harmonic convergences, every planet in the solar system will have a whole number of rotations.

Input

The first line of input consists of a single integer, n ($2 \leq n \leq 10$), giving the number of planets in the enemy solar system. The following line consists of n integers f ($1 \leq f \leq 10$), giving the frequency of each planet. At least two frequencies will be different from each other.

Output

Output a single integer giving the frequency of the harmonic convergence.

input

```
3
6 3 4
```

output

```
12
```

input

```
4
5 4 5 1
```

output

```
20
```

In the first sample input, the harmonic convergence occurs every 12 years. The first planet will have rotated twice, the second planet will have rotated four times, and the third planet will have rotated three times.

I. Intergalactic Transportation

1 second, 256 megabytes

The harmonic convergence was a success! The Robot Empire declared an end to the war and agreed to live peacefully among all the life forms in the galaxy. Out of appreciation for your peacemaking skills, you've been elected Prime Minister of the Galactic Council! Your first act is to build a transportation network across the entire galaxy.

The galaxy is modeled as a 3-D grid of cells, where each side of the grid contains n cells, for a total of n^3 cells in the grid. Your goal is to build exactly $n^3 - 1$ transport links, such that there exists some path between any two cells in the galaxy. A transport link can only be built between adjacent cells. The cell in position (x, y, z) is adjacent to the cells in positions ($x - 1, y, z$), ($x + 1, y, z$), ($x, y - 1, z$), ($x, y + 1, z$), ($x, y, z - 1$), and ($x, y, z + 1$), if these cells exist. Note that cells on the faces, edges, and corners of the grid will only be adjacent to 5, 4, and 3 other cells, respectively.

Each cell belongs to one of the four major alien civilizations: the Keptarks, the Aoriovids, the Vespertence, and the Robot Empire. Constructing a transportation link is an expensive endeavor. The different civilizations each use different technologies, parts of which are incompatible with each other. Thus, the cost of a transport link between cells of two civilizations must be determined from the table below.

- Keptarks + Keptarks -> 2
- Keptarks + Aoriovids -> 9
- Keptarks + Vespertence -> 7
- Keptarks + Robot Empire -> 5
- Aoriovids + Aoriovids -> 3
- Aoriovids + Vespertence -> 8
- Aoriovids + Robot Empire -> 4
- Vespertence + Vespertence -> 1
- Vespertence + Robot Empire -> 10
- Robot Empire + Robot Empire -> 6

Determine the minimum total cost in order to build the transportation network.

Input

The first line of input consists of an integer n $(1 \leq n \leq 50)$, referring to the number of octants per edge. The rest of the input consists of n layers, where each layer consists of n lines of strings of length n, each representing an alien civilization. The 3-D grid is composed of each of the layers in the order and orientation provided. Thus, the cell in position (x, y, z), where $1 \leq x, y, z \leq n$, will be the x_{th} character in the $(y + (z - 1) * n)_{th}$ input string. Characters in the string will be either "K", "A", "V", or "R", which refer to the Keptarks, Aoriovids, Vespertence, and Robot Empire, respectively.

Output

Print a single integer giving the minimum total cost of the transportation network.

input

```
2
KA
VK
RA
VV
```

output

```
28
```

input

```
3
ARR
KVR
KAR
AAV
VRA
KAV
KRA
VKR
RAK
```

output

```
112
```

In the first sample input, the optimal strategy is to link quadrants (1, 1, 1) and (1, 1, 2) for cost 5, quadrants (1, 1, 2) and (2, 1, 2) for cost 4, quadrants (2, 1, 2) and (2, 1, 1) for cost 3, quadrants (1, 1, 1) and (1, 2, 1) for cost 7, quadrants (1, 2, 1) and (1, 2, 2) for cost 1, quadrants (1, 2, 2) and (2, 2, 2) for cost 1, and quadrants (2, 2, 2) and (2, 2, 1) for cost 7. The total cost equals 5+4+3+7+1+1+7 = 28.

J. Joining Spaceships

1 second, 256 megabytes

One year has passed since the end of the great war, and construction of your transportation network has just been completed. In a shocking twist of events, the Robot Empire has reemerged in greater size than ever before. They've hijacked your transportation network in order to invade the entire galaxy at once! It's up to you to stop them.

Unfortunately, the n spaceships in your fleet are in severe need of renovation. They seem to be composed of old parts that have been poorly welded together, and a lot of the engines and thrusters have sustained excessively high levels of teleportation damage. You happen to have n celestial boosters laying around, and you'd like to use one on each of your spaceships. When a regular spaceship with a baseline speed of x is joined with a celestial booster of strength y, the resulting boosted spaceship has a speed of $x * y$. For example, if a spaceship with a baseline speed of 5 is joined with a celestial booster of strength 7, the resulting boosted spaceship would have a speed of 5 * 7 = 35.

Even a boosted spaceship is only useful if it's fast enough. Specifically, you'd like for as many of your boosted spaceships as possible to have a speed greater than or equal to k. Determine how many boosted spaceships can meet this criteria if you join your spaceships and celestial boosters optimally. Note that each celestial booster may only be used on one spaceship.

Input

The first line of input consists of two integers, n and t $(1 \leq n, t \leq 1000)$, representing the number of spaceships and the required speed threshold. The next line of input contains n integers $x_1, x_2, \ldots x_n$, representing the baseline speeds of each spaceship $(1 \leq x_i \leq 1000)$. The next line of input contains n integers $y_1, y_2, \ldots y_n$, representing the strengths of the celestial boosters $(1 \leq y_i \leq 1000)$.

Output

Output a single integer giving the maximum number of boosted spaceships which can have a speed greater than or equal to k, if spaceships are joined with celestial boosters optimally.

input
```
5 20
9 4 8 5 2
3 4 2 3 1
```
output
```
3
```

input
```
3 44
8 5 7
2 1 5
```
output
```
0
```

In the first sample input, an optimal configuration is to match the spaceships of baseline speeds 9, 8, and 5 with the celestial boosters of strengths 3, 3, and 4, respectively. The resulting boosted spaceships will have speeds 27, 24, and 20, respectively, for a total of 3 boosted spaceships of speed greater than or equal to the threshold of 20. In the second sample input, no configuration exists which results in any boosted spaceships meeting the speed threshold.

K. Kingdom Conquest

1 second, 256 megabytes

The construction of your spacefleet is complete! The sheer size and strength of the fleet massively overpowers that of the Robot Empire. You would like to use your overpowered space fleet to conquer the enemy strongholds. Upon visiting a planet, you'll easily be able to conquer it, but difficulty resides in reaching all of the enemy planets. The problem is that your ships are so unnecessarily large that they run out of fuel very quickly. Thus, they can only travel a limited distance before refueling on a planet.

Your fleet starts on your home base at position (0, 0, 0). From your home base, your fleet can reach any planet within a distance of k space units. The distance between two planets is calculated as the Euclidean distance between their positions. Upon visiting a planet, you can conquer it and use it as a base for future travels. For instance, if your fleet can travel a distance of 1, and there are planets at positions (1, 0, 0) and (2, 0, 0), your fleet can visit both planets, since the first planet is a distance of 1 away from your home, and the second planet is a distance of 1 away from the first planet.

Determine the maximum number of enemy planets which can be reached by your fleet.

Input

The first line of input contains two integers, n and k ($1 \leq n, k \leq 1000$), representing the number of planets and the maximum distance your fleet can travel. The next n lines contain three integers, x, y, and z ($0 \leq x, y, z \leq 1000$), representing the coordinates of the planets in 3-D space.

Output

Output a single integer giving the number of planets your fleet can reach.

input
```
6 5
0 3 5
1 8 4
4 0 2
3 3 1
16 6 1
2 12 5
```
output
```
3
```

input
```
5 2
5 1 1
0 2 2
0 0 2
0 0 5
0 0 3
```
output
```
4
```

In the first sample input, the fleet starts at (0, 0, 0) and can reach the planets at (4, 0, 2) and (3, 3, 1), which are separated by distances $2\sqrt{5}$ (approximately 4.47) and $\sqrt{19}$ (approximately 4.36) respectively, both of which are within than the fleet's maximum distance of 5. The fleet can also reach the planet at (0, 3, 5), which is a distance of 5 from the planet at (3, 3, 1), which it can also reach. The fleet cannot reach the planets at (1, 8, 4), (16, 6, 1), or (2, 12, 5) at all, neither from its starting position nor from any of the planets it can reach. The total number of planets which can be reached is 3. In the second sample input, the fleet starts at (0, 0, 0) and can reach the planet at (0, 0, 2). From that position, it can reach planets at (0, 2, 2) and (0, 0, 3). From (0, 0, 3), the fleet can reach the planet at (0, 0, 5). The fleet cannot reach the planet at (2, 1, 1). The total number of planets which can be reached is 4.

L. Lightspeed Cleanup

1 second, 256 megabytes

Excellent job on the planetary conquest! Your space generals are very proud. The strongholds of the Robot Empire have been toppled, and the Galactic Alliance is on the verge of victory. Unfortunately, while you were recklessly dashing around the galaxy, you've left a slew of cosmic pollution in between the planets you traveled through. Although it's a noble task to save the galaxy from tyrannical robot overlords, it's also important to keep the galaxy clean.

There are three different types of cosmic pollution: cosmic dust, cosmic exhaust, and cosmic microparticles. This pollution affects four different types of life forms: cosmic worms, cosmic crawlers, cosmic creepers, and cosmic shadowslitherers. Cosmic worms are quite small and are only affected by cosmic microparticles. Cosmic crawlers are affected by cosmic dust, which prevents them from crawling. Cosmic creepers are affected by cosmic exhaust, which prevents them from creeping. Lastly, cosmic shadowslitherers are affected by all three types of cosmic pollution. But all of this is beside the point. You'd simply like to clean up all of the pollution you've left behind.

Fortunately for you, the cosmic pollution is isolated to only m of the routes you've traveled. Each route spans exactly two of the n planets you've visited, u and v, and takes some w minutes to travel. You'd like to clean up the galaxy as quickly as possible by traveling through these routes with your cleanup fleet. Your cleanup fleet can start at any of the n planets but must end at the same planet. Every route must be traveled at least once. Determine the minimum amount of time needed to clean up the galaxy.

Input

The first line of input contains two integers, n ($1 \leq n \leq 15$) and m ($n-1 \leq m \leq \frac{n(n-1)}{2}$), representing the number of planets and the number of routes. The next m lines contain three integers, u, v, and w ($1 \leq u, v \leq n$, $u \neq v$), ($1 \leq w \leq 1000$), indicating a route between planets u and v that takes w minutes to travel. It will be possible for every route to be traveled.

Output

Output a single integer giving the minimum amount of time it takes to travel all of the routes, given that you must start and end at the same planet.

input

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

output

```
8
```

input

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

output

```
29
```

In the first sample input, the shortest solution is 1->2->3->2->4->1, which takes a total of 8 minutes. Note that the 2->3 link is traversed twice.

M. Moon Mining

1 second, 256 megabytes

Unfortunately, the war has severely depleted the resources of your allies. You've chosen to take up the dangerous endeavor of moon mining. There's an allied starbase at a nearby solar system, and its planets have plenty of moons available to be mined. Each planet is some distance from your starbase. Each moon is some distance from the planet it orbits around. Lastly, your ship has a maximum distance it can travel before breaking down. For safety reasons, your ship must start and end its route at the starbase, potentially traveling to planets and moons throughout its route.

The spaceship must travel in accordance with the following rules.

- The spaceship must start its route at the starbase
- The spaceship may travel between the starbase and a planet
- The spaceship may travel between a planet and one of its corresponding moons
- The spaceship must end its route at the starbase
- The spaceship may NOT directly travel between two planets; it must first return to the starbase
- The spaceship may NOT directly travel between two moons; it must first return to the planet
- The spaceship may NOT travel more than its maximum distance

Determine the maximum total number of moons which can be mined if your route is selected optimally.

Input

The first line of input contains two integers, n, and k, representing the number of planets and the maximum distance the ship can travel ($1 \leq n \leq 50, 1 \leq k \leq 10^{12}$). The second line of input contains n integers $a_1, a_2, \ldots a_n$ ($1 \leq a_i \leq 10^{12}$), representing the distance to each planet. The following n lines of input each describe the moons of a planet. The first integer of the line, m, represents the number of moons ($0 \leq m \leq 50$). The following m integers on the line, $d_1, d_2, \ldots d_m$, describe the distance to each moon ($1 \leq d_i \leq 10^{12}$).

Output

Output a single integer representing the total number of moons which can be mined.

input

```
5 40
5 6 1 9 3
3 2 1 12
3 2 3 8
2 15 3
0
1 21
```

output

```
4
```

input

```
5 11
5 6 1 9 3
3 2 1 12
3 2 3 8
2 15 3
0
1 21
```

output

```
1
```

input

```
5 5
5 6 1 9 3
3 2 1 12
3 2 3 8
2 15 3
0
1 21
```

output

```
0
```

In the first sample input, an optimal route is to travel from the starbase to Planet 1, then from Planet 1 to its moon at distance 2, then from its moon back to Planet 1, then from Planet 1 to its moon at distance 1, then from its moon back to Planet 1, then from Planet 1 back to the starbase, then from the starbase to Planet 2, then from Planet 2 to its moon at distance 2, then from its moon back to Planet 2, then from Planet 2 to its moon at distance 3, then from its moon back to Planet 2, then from Planet 2 back to the starbase. The total distance traveled is 5+2+2+1+1+5+6+2+2+3+3+6=38, which is within the ship's maximum distance of 40. This route allows 4 moons to be mined. In the second sample input, the optimal route is to travel from the starbase to Planet 3, then from Planet 3 to its moon at distance 3, then from its moon back to Planet 3, then from Planet 3 back to the starbase. The total distance traveled is 1+3+3+1=8, which is within the ship's maximum distance of 11. This route allows 1 moon to be mined. In the third sample input, there is no possible route which travels to a planet, to a moon, back to the planet, and back to the starbase within the maximum distance of 5. Thus, no moons are mined.

N. Nuclear Reactor

1 second, 256 megabytes

The moon mining mission was a great success! You've collected lots of nuclear fuel for use in your mysterious superweapon. Your space engineers are still working out the details of the weapon, so they've given you the mission-critical task of weighing the amount of fuel you've mined.

You have some number of fuel blocks, each of which is a cube with a length of exactly one meter. Weighing the fuel blocks is a challenging task, because the fuel blocks are somewhat difficult for you to lift on account of how large and heavy they are. Luckily, you're sort of a math genius, and you can calculate the weights of all the fuel blocks without actually having to lift them up.

Fuel blocks are composed of either uranium, plutonium, or thorium. A uranium fuel block weighs 19 tons, a plutonium fuel block weighs 20 tons, and a thorium fuel block weighs 12 tons. You would like to determine the combined weight, in tons, of all of the fuel blocks.

Input

The first line of input consists of an integer n $(1 \leq n \leq 100)$, giving the number of fuel blocks. The following n lines each represent a type of fuel block by consisting of one of the following strings: "uranium", "plutonium", or "thorium".

Output

Output a single integer giving the total combined weight of all the fuel blocks.

input
```
5
thorium
plutonium
thorium
uranium
uranium
```
output
```
82
```

input
```
1
plutonium
```
output
```
20
```

In the first sample input, the combined weight equals 12 + 20 + 12 + 19 + 19 = 82.

O. Oblique Perimeter

1 second, 256 megabytes

While you were kept busy with the vital task of weighing your fuel blocks, your space engineers have finished construction of the superweapon. The only problem is that it still needs time to charge. While your superweapon is charging, the Robot Empire is preparing an attack that would completely destroy the weapon, along with all hope for the galaxy. Perhaps there can be some way of protecting against their attack.

The space architect has proposed a hexagonal barricade system, where each barricade is a horizontal wall of unit length. Barricades connect with each other end-to-end at a precise angle of 120 degrees, thus forming an overall structure of one or more adjacent hexagons. The space architect was nice enough to go ahead and design the barricade system without consulting your advice, and he probably didn't do a very good job. You're concerned that his design does not enclose a large enough area to store the superweapon.

You would like to determine the total number of hexagons enclosed within the barricade system.

Input

Input consists of a single string of length z $(6 \leq z \leq 1000)$. Each character in the input string is either "L" or "R", representing a left or right turn with respect to the previous barricade in the sequence. It is guaranteed that the barricade system forms a closed, non-intersecting loop.

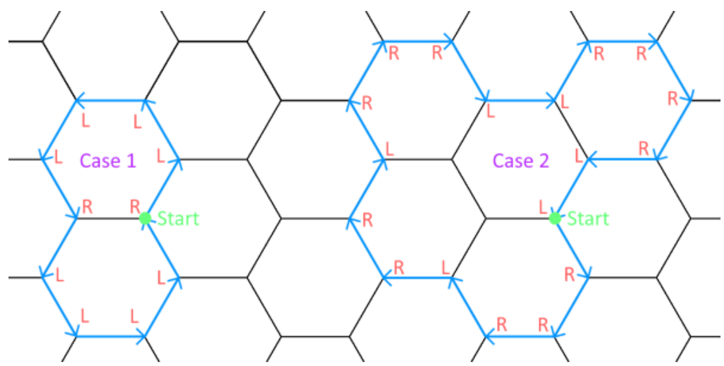
Output

Output a single integer giving the total number of hexagons enclosed in the barricade system.

input
```
RLLLLRLLLL
```
output
```
2
```

input
```
LRRRLRRLRRRLLRRRRL
```
output
```
5
```

The image below shows the areas enclosed by the first (left) and second (right) sample inputs. The starting point is shown in green, the barricades are shown in blue, and the turns are shown in red. In the first sample input, the barricades enclose a total area of 2 hexagons. In the second sample input, the barricades enclose a total area of 5 hexagons.



P. Pocket Universe

1 second, 256 megabytes

Over the course of your space adventure, you've leveraged your programming skills to build and optimize spaceships, command your own space legion, forge alliances with aliens, and lead the Galactic Council against its greatest threat ever encountered. Thanks to your courageous efforts, the menacing Robot Empire has at last been contained to a single solar system. With the resources gathered from your calculated mining expedition, you can finally activate your superweapon: the Interdimensional Shifter.

The target solar system currently resides in Dimension 0. Your goal is to shift the solar system x dimensions, such that it will be locked away in a pocket universe. You'll be using the Interdimensional Shifter to reach your goal and would like to consume as few fuel blocks as necessary.

The Interdimensional Shifter has three modes: the Unit Shifter, the Forward Shifter, and the Backward Shifter. The default mode is the Unit Shifter. Swapping between modes requires one fuel block. Executing a shift also requires one fuel block. If you execute a shift while set to the Unit Shifter, the target will be shifted 1 dimension forward. If you execute a shift while set to the Forward Shifter, the target will be shifted f dimensions forward. Lastly, if you execute a shift while set to the Backward Shifter, the target will be shifted b dimensions backward.

For example, if $x = 14$, $f = 9$, and $b = 5$, the optimal strategy is as follows. First, the Unit Shifter is used once, to shift the target to Dimension 1, consuming one fuel block. Then, the mode is switched to the Forward Shifter, consuming one fuel block. The Forward Shifter is then used twice, shifting the target to Dimension 19 ($1 + 9 + 9 = 19$) and consuming two fuel blocks. Then, the mode is switched to the Backward Shifter, consuming one fuel block. Lastly, the Backward Shifter is used once, shifting the target to Dimension 14 ($19 - 5 = 14$). The total number of fuel blocks consumed is 6, which is the minimum number of fuel blocks possible in order to reach the desired dimension.

Input

Input consists of one line containing three integers, x, f, and b ($1 \leq x, f, b \leq 10^5$), representing the target dimension, the strength of the forward shifter, and the strength of the backward shifter.

Output

Print a single integer giving the minimum number of fuel blocks which need to be used in order to shift the enemy solar system to the target dimension.

input
```
14 6 2
```
output
```
5
```

input
```
25 13 9
```
output
```
9
```

In the first test case, the optimal strategy is to use the Unit Shifter twice, swap to the Forward Shifter, then use the Forward Shifter twice, for a total cost of 5 fuel blocks. in the second test case, the optimal strategy is to swap to the Forward Shifter, use the Forward Shifter four times, swap to the Backward Shifter, then use the Backward Shifter three times, for a total cost of 9 fuel blocks.