Maratona de Programação da SBC 2019

This problem set is used in simultaneous contests: Maratona de Programação da SBC 2019
Tercera Fecha Gran Premio de México 2019
Tercera Fecha Gran Premio de Centroamérica 2019

September 14th, 2019

Problems book

General Information

This problem set contains 13 problems; pages are numbered from 1 to 17, Without considering this page. Please, verify your book is complete.

A) Program name
1) Solutions written in C/C++ and Python, the filename of the source code is not significant, can be any name.
2) Solutions written in Java, filename should be: *problem_code*`.java` where *problem_code* is the uppercase letter that identifies the problem. Remember in Java the main class name and the filename must be the same.
3) Solutions written in Kotlin, filename should be: *problem_code*`.kt` where *problem_code* is the uppercase letter that identifies the problem. Remember in Kotlin the main class name and the filename must be the same.

B) Input
1) The input must be read from *standard input*.
2) The input is described using a number of lines that depends on the problem. No extra data appear in the input.
3) When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input.
4) Every line, including the last one, ends with an end-of-line mark.
5) The end of the input matches the end of file.

C) Output
1) The output must be written to *standard output*.
2) When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output.
3) Every line, including the last one, must end with an end-of-line.

Promo:

Sociedade Brasileira de Computação

v1.0

Problem A
Artwork

The Mona Dura is one of the most valuable artworks in Nlogonia Museum. The famous painting is displayed in a rectangular room of M by N meters. The room entrance is in a corner of it, while the Mona is in the corner diagonally opposite to the entrance.

To prevent theft, the room has motion sensors that are activated every night when the museum closes. Each sensor has a sensitivity S, such that the sensor triggers an alarm if it detects any movement at no more than S meters from its location.

Tonight a thief broke into the museum with the purpose to steal the Mona Dura. To achieve his goal, the thief needs to enter the room and reach the painting without being detected by any of the motion sensors, that is, he must keep a distance longer to S_i meters from the i-th motion sensor all the time, for all the sensors.

The thief has access to the plants of the museum, therefore, he knows the size of the room, the coordinates, and the sensitivities of each of the motion sensors. Given this information, your task is to determine if it is possible for the thief to steal the Mona Dura.

Input

The first line of input contains three integer numbers, M, N, and K, representing the size of the room, and the number of sensors, respectively. ($10 \leq M, N \leq 10^4$, $1 \leq K \leq 1000$). The entrance to the room is located at position $(0, 0)$, and the painting at position (M, N).

Each of the next K lines describes one of the K sensors, it contains three integer numbers, X, Y, and S, where (X, Y) represents the sensors location and S represents the sensor's sensitivity. ($0 < X < M$, $0 < Y < N$, $0 < S \leq 10^4$). All dimensions and coordinates in the input are in meters. It is guaranteed that all sensors have different coordinates.

Output

Your program must output a single line containing the character '`S`' in case the painting can be stolen, or the character '`N`' otherwise.

Input example 1	**Output example 1**

`10 22 2`
`4 6 5`
`6 16 5`
	`S`

Input example 2	**Output example 2**

`10 10 2`
`3 7 4`
`5 4 4`
	`N`

Input example 3	**Output example 3**

`100 100 3`
`40 50 30`
`5 90 50`
`90 10 5`
	`S`

Problem B
Buffoon

The Kingdom of Matchings is governed by a generous commander. The commander's fame and great qualities are known to all, including neighboring kingdoms. One of his most famous qualities is his good humor, which is nourished daily by a court buffoon, elected annually at the Great Comedy Contest (GCC) of the kingdom. The court buffoon helps to relieve all the tension of the various political meetings the work demands, rejoicing not only the commander but the whole kingdom.

Young Carlos is a great comedian whose dream is to become next season's buffoon. He has spent the past few months writing new jokes and puns of various kinds, many of which are about his own (tiny) stature. The time has come for the buffoon election and a total of N candidates have registered. Each candidate will have five minutes to perform in front of the audience. After the performances, each citizen of the Kingdom of Matchings may vote for one of the candidates, and the most voted candidate will be elected as court buffoon. If there is a tie between one or more candidates, the one who registered first is elected. Knowing this, young Carlos spent nights in front of the electoral office and ensured that his application was the first to be registered.

After the votes, it remains only to determine the results. The voting machine generates a report with N integers, corresponding to the number of votes for each candidate, ordered in order of registration. Your mission is to determine if young Carlos was elected or not.

Input

The first line of input contains an integer N, ($2 \leq N \leq 10^4$). The next N lines will contain N positive integers $v_1, \ldots, v_N$, one on each line, corresponding to the number of votes each candidate received, in order of registration. Since the Kingdom of matchings population is 100,000 people, the total number of votes will not exceed this value, i.e $\sum_{i=1}^{N} v_i \leq 100,000$.

Output

Your program must output a single line containing the character '`S`' if young Carlos is elected as buffoon, or the character '`N`' otherwise.

Input example 1	**Output example 1**
3 1000 1000 1000	S

Input example 2	**Output example 2**
5 1 2 3 4 5	N

Problem C
Crossings With Danger

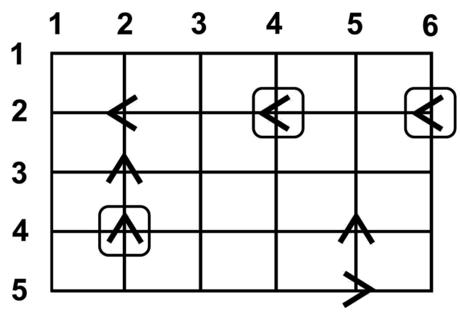
The Agency for Simulation of Vehicles (ASV), managed by its founder Crishna, is working on a project to retrieve data related to vehicle collisions in crossings of some regions in the country.

The first simulation the ASV will perform consists in randomly place C vehicles in crossings of a well determined region. At the beginning of the simulation there will be a maximum of one vehicle at each crossing. A region is represented by N horizontal lanes that cross with M vertical lanes.

Once the simulation is started, each vehicle will move following its initial direction (Nort, South, East, or West) at a constant velocity of 1 crossing per second.

If two or more vehicles get to the same crossing at the same time, the vehicles will collide and stop moving. If another vehicle goes through a crossing in which a collision ocurred previously then the vehicle will collide with the stopped vehicles at the crossing. For unknown reasons, when two vehicles collide horizontally **between** two crossings, both will stop at the east crossing, and if two vehicles collide vertically **between** two crossings, both will stop at the north crossing

In the picture below a simulation is shown where $N = 5$, $M = 6$, and $C = 7$, vehicles are represented by arrows indicating their initial directions. Note that the 3 highlighted vehicles will collide in some moment.



Since the ASV simulation software is not yet completely reliable, they will provide examples of initial configurations to count the number of vehicles that will not collide.

Input

The first line of input contains three integer numbers N, M, and C, ($2 \leq N, M \leq 10^5$ and $1 \leq C \leq min(10^5, N \times M)$) representing the amount of horizontal lanes, the amount of vertical lanes, and the amount of vehicles in the simulation, respectively. Each of the following C lines will contain two integers A_i and B_i and one character D ($1 \leq A_i \leq N$ and $1 \leq B_i \leq M$) representing that the i-th vehicle is initially at the crossing between the A_i-th horizontal lane and the B_i-th vertical lane, with a direction indicated by D: '`N`' for the North, '`S`' for the South, '`L`' for the East, and '`O`' for the West.

Output

Your program must output a single line, containing an integer number, representing the number of vehicles that will not collide.

Input example 1	**Output example 1**

`5 6 7`
`2 2 O`
`3 2 N`
`4 2 N`
`4 5 N`
`2 6 O`
`5 5 L`
`2 4 O`
	`4`

Input example 2	**Output example 2**

`2 2 3`
`1 1 L`
`1 2 O`
`2 2 N`
	`0`

Input example 3	**Output example 3**

`2 2 3`
`1 1 L`
`1 2 O`
`2 1 N`
	`1`

Problem D

Denouncing Mafia

Nlogonia police department is investigating the local mafia. They already know all the members and the structure of the organization: The Nlogonian mafia has N members, and each is identified by an integer between 1 and N, where 1 is the number that identifies the mafia boss. In addition, every member except the boss is a direct subordinate of another member.

Even after months of investigation, police still do not have enough information to arrest any mafia members for any crime. So they decided to ask for the help of a seer: given a mafia member, the seer can magically guess the crimes he has committed, and the police can then confirm them by interrogation.

In addition, when a Nlogonian mafia member is interrogated, he not only admits his crimes, but also, in exchange for a lighter sentence, he reports the crimes of his direct superior. If the superior has not already been arrested, the police can interrogate him as well, and he will then report his superior, and so on, until they reach the boss.

Unfortunately, the seer has only enough energy to guess a maximum of K mafia members, and the police wants to use the seer powers carefully so they can arrest as many bad guys as possible. Given K and the full structure of the mafia, what is the maximum amount of mafia members the police can arrest?

Input

The first line contains two integers, N and K, where N is the number of mafia members and K is the maximum number of mafia members the seer can guess ($3 \leq N \leq 10^5$, $1 \leq K < N$). The second line contains $N - 1$ integers, where the i-th of them is the number that identifies the direct superior of the mafia member with identifier $i + 1$. All integers in the second line of input are guaranteed to be between 1 and N, and all mafia members are subordinates of the boss, directly or indirectly.

Output

Your program must output a single line, containing an integer, representing the maximum number of mafia members the police can arrest.

Input example 1	Output example 1
`8 2` `1 1 2 3 4 4 6`	`7`

Input example 2	Output example 2
`10 3` `1 1 2 2 3 3 4 4 5`	`8`

Problem E
Exhibition of Clownfish

The Great Nlogonia Aquarium welcomes thousands of visitors every month. One of its most popular attractions is the exhibition of clownfish, a hall with several tanks of shoals of this beautiful white and orange species. Visitors have the opportunity to learn many curiosities about clownfish, including their social organization: shoals of clownfish are led by females, and when the last female dies or leaves the shoal, one of the remaining males mutates, becomes female and now leads the shoal!

As soon as he learned this, Zelius the Janitor decided to play a trick on the Aquarium and make all the clownfish in the exhibit go female! For this, he will move the fish from one tank to another at night when the Aquarium is closed. If by the end of the night any tank is left with one or more males and no female, the next day one of them will have become a female.

Not to arouse suspicion from the other staff, Zelius can only move one clownfish each night, and each fish can only be moved between the exhibition's tanks. Each tank is large enough to hold an unlimited amount of fish, and Zelius can leave as many empty tanks as he wants. We can assume that no one else will touch the fish, and that no fish will be born, die, added or removed from the tanks.

Zelius counted how many male and female fish currently live in each tank of the exhibit. Now he needs your help to plan his movements to turn all clown fish into females in the shortest possible time.

Input

The first line of input contains a single integer N, the number of tanks in the exhibit ($2 \leq N \leq 3000$). Each of the following N lines corresponds to one of the tanks and contains two integers, M and F, the amounts of male and female fish in that tank, respectively ($0 \leq M, F \leq 10^5$, $M = 0$ or $F > 0$).

Output

Your program must output a single line, containing an integer, representing the minimum number of movements required.

Input example 1	Output example 1
2 2 1 0 2	2

Input example 2	Output example 2
2 2 5 1 3	7

Input example 3	Output example 3
4 2 3 0 0 3 1 0 0	5

Problem F
Forests in Danger

Due to the advanced deforestation in the last decades, the rivers of Nlogonia registered a significant reduction in their flow rate. As Nlogonia is a developed nation that bases its decisions on technical data, the nation's leader has commissioned a series of studies to understand what steps should be taken to ensure water for future generations.

The technical report prepared by the scientists involved in the project was categorical: a percentage of the country's territory must have its vegetation conserved. More than that, the areas near the rivers must be the most preserved.

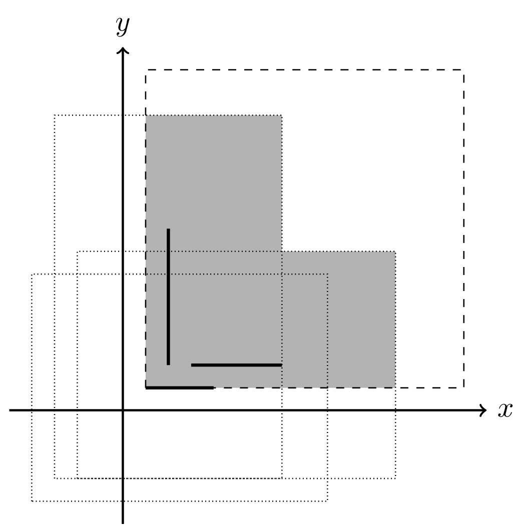
New environmental legislation will come into force, in which areas up to a distance from the river banks will be part of the preservation area. The ideal value of this distance is still unknown, but the technical report has already determined the percentage of the nation's territory that needs to be preserved.

Given your technical capabilities, you have been sought to help determine the distance around the rivers that must be preserved in order to achieve the required percentage of conserved area.

The rivers of Nlogonia can be represented in the plane as straight segments parallel to the axes. Given a distance r, the area of the territory to be preserved is determined as follows: For each river, the surrounding preserved area corresponds to the smallest rectangle containing the segment representing the river, leaving a minimum distance of r units between any point in the segment and any point outside the rectangle. The territory of Nlogonia is defined as a rectangle with sides parallel to the axes, so that every river is parallel to some border of the territory.

Given an integer P between 1 and 100, you must determine the smallest integer value for r that guarantees the preservation of $P\%$ of Nlogonia territory.

The figure below illustrates the first example of the input. The territory of Nlogonia is represented by the dashed region, and the preserved area is represented by the gray region:



Input

The first line contains an integer N ($1 \le N \le 10^4$) indicating the number of line segments representing the rivers of Nlogonia. Each of the following N lines contains 4 integers: x_1, y_1, x_2, and y_2, where (x_1, y_1) and (x_2, y_2) are the ends of a straight line representing a river. Since the rivers of Nlogonia are parallel to the borders, it is guaranteed that $x_1 = x_2$ or $y_1 = y_2$.

The next line contains an integer P ($1 \le P \le 100$) indicating the minimum percentage of territory that must be preserved. The last line contains 4 integers x_1, y_1, x_2, y_2, where (x_1, y_1) is the lower left corner and (x_2, y_2) is the upper right corner of the rectangle, which represents the territory of Nlogonia, with sides parallel to the coordinated axes.

Each coordinate described in the input is an integer between 0 and 10^5. You can assume that all rivers are fully contained in Nlogonia territory.

Output

Your program must output a single line, containing an integer number r, representing the minimum integer value that can be set for r to guarantee the preservation of $P\%$ of Nlogonia territory.

Input example 1	Output example 1
3 1 1 4 1 2 2 2 8 3 2 7 2 50 1 1 15 15	5

Input example 2	Output example 2
1 0 0 0 4 50 0 0 4 4	2

Problem G
Getting Confidence

Sicrana loves ornaments. At home, she has N ornaments that are displayed lined up on a large shelf. Each ornament is identified by a distinct integer between 1 and N.

One day while playing ball indoors, Fulano, Sicrana's son, hit his mother's ornament shelf with the ball, knocking all the ornaments to the floor. Luckily no ornament was damaged with the fall. If Fulano can put all the ornaments back on the shelf exactly as they once were, his mother may not realize that something wrong has happened.

Since Fulano has a bad memory, he can't remember the order the ornaments were originally in, so he needs your help. For each ornament i, Fulano will tell you N numbers between 1 and 100, where the j -th value indicates the confidence Fulano has that the i ornament was originally in the j position on the shelf. To maximize Fulano's confidence that he won't get grounded, when choosing an order and placing the decorations, Fulano will multiply the confidence that each ornament is in the right place. More formally, Fulano's total confidence in a given order of the decorations is calculated as follows: If p_i is the position occupied by the i -th ornament and $a(i, j)$ is Fulano's confidence that the i ornament was originally in the position j, then the Fulano's total confidence is given by $\prod_{i=1}^{N} a(i, p_i)$.

Since there are so many different ways for positioning the ornaments, your mission, if you accept it, is to find the order that maximizes Fulano's total confidence.

Input

The first line of input contains an integer N ($1 \leq N \leq 100$), indicating the number of ornaments. Each of the following N lines contains N integers between 1 and 100. The j-th value in the i-the line represents the confidence Fulano has that the i ornament was originally in the j position on the shelf.

Output

Your program must output a single line, containing N integers, representing the order in which Fulano should place the ornaments to maximize his total confidence. If there is more than one order that results in maximum confidence, either order will be accepted.

Input example 1	**Output example 1**
3 1 15 37 42 8 25 77 2 1	3 1 2

Input example 2	**Output example 2**
2 15 1 33 42	1 2

Problem H
Hour for a Run

Vinicius takes his fitness very seriously, and every morning at 6 a.m., rain or shine, in summer and winter, he runs in a track around a pond. Along the race track there are N equally spaced signs. To not be discouraged from exercising, Vinicius counts the number of signs he has passed and checks to see if he has run at least 10%, at least 20%, ..., at least 90% of his training.

Let's help Vinicius by calculating for him the number of signs he needs to count to have completed at least 10%, 20%, ..., 90% of his training, given the number of laps he wants to run and the total number of signs along the track.

For example, suppose Vinicius wants to run 3 laps and the track has 17 signs. To ensure that he has run at least 30% of his training, he needs to count 16 signs. To guarantee at least 60%, he needs to count 31 signs.

Input

The input consists of a single line, which contains two integers, V and N ($1 \leq V, N \leq 10^4$), where V is the desired number of laps and N is the number of signs along the track.

Output

Your program must output a single line, containing nine integers, representing the numbers of signs that must be counted to ensure that at least 10%, 20%, ..., 90% of the training has been completed, respectively.

Input example 1	Output example 1
`3 17`	`6 11 16 21 26 31 36 41 46`

Input example 2	Output example 2
`5 17`	`9 17 26 34 43 51 60 68 77`

Input example 3	Output example 3
`3 11`	`4 7 10 14 17 20 24 27 30`

Problem I
Interplanetary

It is year 2306 and with the advancement of nanotechnology, interplanetary travel is becoming generally available. Bibika works at the largest interplanetary travel agency in the universe and receives interested clients every day.

Bibika customers are demanding and make several constraints before closing their travel itinerary, such as minimizing the total distance traveled. But the biggest constraints are on the temperatures of the planets visited on the route (excluding the source and destination planets). The temperature of a planet, measured in degrees Anidos, can range from 10^9 negative degrees Anidos to 10^9 positive degrees Anidos. Bibika's clients come from planets of varying climates and, consequently, have different temperature preferences: some worry about very cold planets and others about very hot planets. Bibika needs to plan the travel route so as to save its customers from any discomfort, even if the total length of the route is not as short as possible (or even there is no route: in which case Bibika simply informs customers that the trip is impossible).

Bibika provided you with the historical average temperature of each of the N planets and the R routes that connect pairs of planets directly (it is guaranteed that between two planets there is at most one direct route), along with their respective distances. She will also provide you with travel requests from Q customers. Each travel request consists of a source planet A, a destination planet B, and the customer's restriction on intermediate planet temperatures: each customer may require only planets with temperatures between the lowest K or the highest K temperatures among all N planets.

Your task is, for each travel request, to find the shortest possible distance given the restrictions described, or to say that such travel is impossible.

Input

The first line of input contains two integers N and R ($2 \leq N \leq 400$ and $0 \leq R \leq N \cdot (N-1)/2$), which represent the number of known planets and number of direct routes between them. The first planet is represented by the number 1, the second by the number 2, ..., up to N-th represented by the number N. The second line of input contains N integers T_i ($-10^9 \leq T_i \leq 10^9$), which represents the average temperature of each of the planets. Then there will be R lines, each with three integers X, Y and D ($1 \leq X, Y \leq N$ where $X \neq Y$ and $1 \leq D \leq 10^3$), which represents a direct route of length D between planets X and Y. Then there will be an integer Q ($1 \leq Q \leq 10^5$), which represents the number of customer travel orders. Finally, each of the following Q lines will contain four integers A, B, K and T ($1 \leq A, B, K \leq N$ with $A \neq B$ and $T \in \{0, 1\}$), which represents a customer who wants to go from planet A to planet B going only through planets with temperatures that are among the coldest K temperatures if $T = 0$ or the hottest K temperatures if $T = 1$.

Output

Your program must print one line per customer request, containing an integer representing the shortest total travel distance between the two planets given the client's restrictions, or -1 if the trip is impossible.

Input example 1	Output example 1
7 9 -53 -180 456 420 -210 15 150 1 2 2 1 3 1 2 3 4 2 4 2 2 5 5 3 4 6 6 4 10 4 5 4 3 7 2 4 1 5 2 1 1 2 1 1 5 6 1 0 1 7 2 1	11 2 -1 3

Input example 2	Output example 2
6 5 5 10 20 10 10 8 1 2 5 2 3 5 3 4 5 4 5 5 5 6 5 4 1 6 2 1 1 6 1 1 4 5 1 0 2 4 1 1	25 -1 5 10

Problem J
Jar of Water Game

The schedule of the day for a programming contest usually follows the same pattern: warmup in the morning, followed by lunch time, some rest time, final adjustments of the competition environment and then the start of the contest.

At rest time some contestants prefer to relax, others prefer to socialize and some have a habit of playing some game of cards. Luciano and his friends enjoy playing a game known as “Jar of Water”. Tired of not being the winner, Luciano wants to write a program that, given the initial cards of all players (don’t ask me how he knows it), determines whether he will win or not. If he is not going to win, he can then make up any excuse and ask not to participate in that round.

The game works as follows:

- The deck used has the cards: “A23456789DQJK” (in this order, from lowest to highest value), where suits are ignored. In addition, the deck has one more extra card: a wildcard.
- N contestants sit side by side in a circle. Contestant 1 is immediately to the left of contestant 2, which is immediately to the left of contestant 3, and so on until completing the circle with the N-th contestant immediately to the left of contestant 1. A contestant K is selected randomly to start the game.
- In a game with N contestants there will be four cards of each of N different values and one wildcard. At the beginning of the game, the contestant K receives the wildcard; The remaining cards are shuffled and distributed among the players so that each player receives four of them.
- In each round, the contestant in turn chooses one of his cards and passes it to the contestant to his right. The contestant who received the card will be the next contestant in turn.
- We say a contestant is in a winning state if he has exactly four cards in hand and they all have the same value. The game ends as soon as at least one competitor is in the winning state. In this case, the competitor with lowest number in winning state will be declared the winner of the game.

The card that will be passed from one competitor to the next is selected by the following rule:

- The wildcard can never be passed as soon after being received. This also applies to the starting player who received the card dealer’s wildcard just before the first round.
- The contestant will, whenever possible, pass the wildcard to the next.
- If the contestant can not pass the wildcard, the contestant will choose the card that appears least in his hand and move that card to the next competitor. If there is more than one card that appears less often, the contestant will select, among these, the one with lowest value in the order described above.

Knowing the rules, help Luciano write a program that, given the initial setup of the game, tells which player will be declared the winner of the game.

Input

The first line of input contains two integer numbers N and K ($2 \leq N \leq 13$ e $1 \leq K \leq N$) representing the number of contestants in the game and the contestant who will start the game, respectively. Each of the following N lines contains four characters, representing the initial cards for the i-th contestant (not considering the wildcard).

Output

Your program must output a single line, containing an integer number, representing the contestant who will be declared winner of the game.

Input example 1	Output example 1

`2 1`
`33J3`
`JJJ3`
	`2`

Input example 2	Output example 2

`2 2`
`A2A2`
`22AA`
	`2`

Input example 3	Output example 3

`4 2`
`774Q`
`JJQ7`
`44Q7`
`4QJJ`
	`3`

Input example 4	Output example 4

`3 1`
`JQAA`
`JJJA`
`QQQA`
	`3`

Problem K
Keep Calm and Sell Balloons

Walter sells balloons from door to door. Every day he chooses a street in his city and visits every house there, offering his colorful balloons.

Each street in the city of Walter has the same number of houses on both sides, and all the houses in the city are the same size. That way, each street can be viewed as a $2 \times N$ matrix, where each cell is a house, and N is the amount of houses along each side of the street.

After choosing the street of the day, Walter visits each house on that street exactly once. He can start his way in any house, but can only move between adjacent houses horizontally, vertically or diagonally.

1	2	3	4	5	6
7	8	9	10	11	12

The table above illustrates an example where $N = 6$. After visiting house number 1, Walter could only proceed immediately to houses number 2, 7, and 8 (that is, if he has not visited them before). And after visiting house number 11, the next house on the way could only be one of 4, 5, 6, 10, or 12.

Today, before leaving home, Walter looked at the city map to count the number of houses on either side of the chosen street. Now he wants to know how many different ways he can visit all the $2N$ houses on the street, following the rules outlined. Two ways to visit the houses are different if and only if the order of the houses varies: that is, if there are two houses A and B such that A is visited before B in one order and B is visited before A in the other.

Input

The input consists of a single line, which contains an integer N ($1 \leq N \leq 10^9$).

Output

Your program must output a single line, containing an integer, representing the number of differents ways to visit all houses in the street. Since this number can be very big, print the remainder of dividing it by $10^9 + 7$.

Input example 1	**Output example 1**
`2`	`24`

Input example 2	**Output example 2**
`3`	`96`

Input example 3	**Output example 3**
`4`	`416`

Input example 4	**Output example 4**
`61728`	`654783381`

Problem L
Less Coin Tosses

Carla and Daniel have decided to play heads or tails to decide who will wash the dishes today. They will play with one of the old coins from Carla's collection. This makes Daniel worried, because these coins are crooked and unbalanced: when tossing a coin, the odds of getting heads and tails are not necessarily equal.

Carla knows her coins well, and can choose one that maximizes her chances of winning. Therefore, Daniel devised a scheme to make the game completely fair, regardless of the chosen coin. First, each will be assigned a nonempty set of binary strings of size N. No string can belong to both, and some strings may not be included in either set. For example, for $N = 3$, a valid way to distribute the strings would be:

- "010" and "110" for Carla;
- "001" and "011" for Daniel;
- "000", "100", "101" and "111" for neither.

After distributing the strings, Carla and Daniel will toss the same coin N times and write down the sequence of results, where each head equals 0 and each tail equals 1. If the resulting binary string belongs to Carla's set, she is the winner. If it belongs to Daniel's set, he is the winner. If the string does not belong to either of them, the coin is tossed another N times to generate a new string. The process is repeated as many times as necessary until they get a winner.

The proper functioning of this scheme depends on the distribution of the strings between Carla and Daniel: the probability of generating a string of the Carla's set must be equal to the probability of generating a string of Daniel's set. In other words, let $P(S)$ be the probability that a binary string S of length N will be generated by a sequence of N tosses of the same coin, possibly unbalanced. The total of P for all strings in Carla's set must be the same as the total of P for all the strings in Daniel's set.

In addition to distributing the strings fairly, Carla and Daniel want to avoid having to repeat the coin tosses as much as possible, so they want to minimize the number of strings that do not belong to either set. Given N, determine the minimum possible number of unassigned binary strings.

Input

The input consists of a single line, which contains contains an integer number N, the number of coin tosses and binary strings size ($2 \leq N \leq 10^{18}$).

Output

Your program must output a single line, containing an integer, representing the minimum possible number of unassigned binary strings.

Input example 1	**Output example 1**
3	4

Input example 2	**Output example 2**
5	4

Input example 3	**Output example 3**
8	2

Problem M
Maratona Brasileira de Popcorn

The "Maratona Brasileira de Popcorn" is a competition that takes place annually to find out which team is the most organized, prepared and well-trained in the art of eating popcorn. It is organized by Brazilian Society of Popcorn Eaters (SBCp, its acronym in Portuguese), which periodically meets to discuss the rules and format of the competition.

The competition consists of N popcorn bags placed side by side, where each bag has an arbitrary amount of popcorn. For added fun, the competition takes place in teams, each made up of C competitors. Since the "Maratona Brasileira de Popcorn" is a serious event that values, above all, the health of the competitors, the medical commission has imposed that each competitor may eat a maximum of T popcorn per second to avoid possible sickness.

At its last meeting, SBCp defined two new rules for the 2019 edition:

- Each team competitor must eat a contiguous sequence of popcorn bags. It is perfectly valid that a competitor does not eat any popcorn.
- All popcorn in the same bag must be eaten by a single competitor.

The goal of the competition is to eat all the popcorn in the shortest possible time as the C competitors can eat in parallel and they will abide by all rules imposed by the SBCp.

Input

The first line of input contains three integer numbers N, C y T ($1 \leq N \leq 10^5$, $1 \leq C \leq 10^5$ and $1 \leq T \leq 50$), representing the number of popcorn bags in the competition, the number of competitors in the team, and the maximum amount of popcorn per second a competitor can eat. The second line contains N integers P_i ($1 \leq P_i \leq 10^4$), representing the amount of popcorn on each of the N popcorn bags.

Output

Your program must output a single line, containing an integer number, representing The minimum amount of seconds it takes for the team to eat all the popcorn if they organize themselves as best possible.

Input example 1	**Output example 1**
`5 3 4` `5 8 3 10 7`	`4`

Input example 2	**Output example 2**
`3 2 1` `1 5 1`	`6`

Input example 3	**Output example 3**
`3 2 1` `1 1 5`	`5`