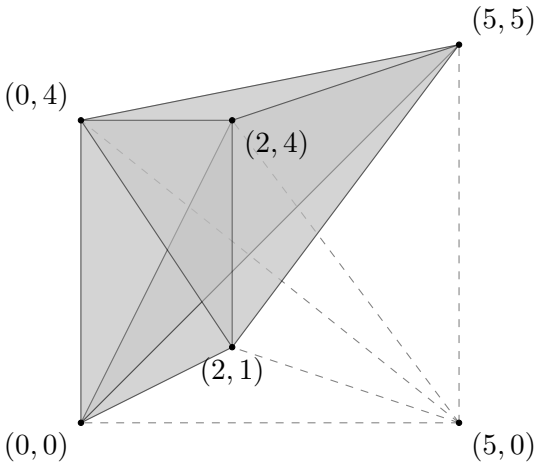
A. Antenna (antenna)

Problem Statement

PC Telecommunication Corporation is a newly founded company and they have planned to build N antennae in Taiwan. If we describe the map of Taiwan with 2D coordinates, the i-th antenna will be built at position (x_i, y_i). These antennae are capable of communicating with low-energy electromagnetic waves which have less impact on surrounding dwellers' health. However, in order to connect with these antennae, one must be inside some triangle formed by three antennae. That is, the service range of these antennae is defined by the union of every triangle formed by three different antennae.

However, Taiwan is a place long prone to typhoons. Therefore PC telecom corp. is worrying about the potential losses brought about by these typhoons. Take the following figure as an example. The area of service range of the six antennae is 22.5, but if the antenna located at $(5, 0)$ becomes out of service, the service range will be reduced to 12.5.



Although PC telecom corp. has updated their architectural technology, their risk management division still wants to know that, for a given set of N antennae, if one of them is unavailable due to natural disasters, what is the minimum area of the service range among all posibilities?

Since the area multiplied by 2 is always an integer, you should output the area multiplied by 2.

Input Format

N
x_1 y_1
x_2 y_2
$\vdots$
x_N y_N

- N is the number of antennae.
- (x_i, y_i) is the position of i-th antenna.

Output Format

$2A$

- A represents the minimum service area of the rest $N-1$ antennae. Note that you should output A multiplied by 2.

Constraints

- $2 \leq N \leq 5 \times 10^5$
- $0 \leq x_i, y_i \leq 10^9$
- $(x_i, y_i) \neq (x_j, y_j) \quad (i \neq j)$
- All inputs are integers.

Examples

Sample Input	Sample Output
2 4 5 6 7	0
6 0 0 0 4 5 0 5 5 2 1 2 4	25
10 1 13 11 11 7 6 11 12 13 1 1 6 6 13 3 12 12 0 10 4	137

Scoring

In this problem, there are two subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Score	Additional Constraints
1	24	$N \leq 3000$
2	76	No additional constraints



This page is intentionally left blank.

B. Bubble Tea (bubbletea)

Problem Statement

Nathan has just spent the last 24 hours fixing the workstation. As a Taiwanese, he decides to have some bubble tea to relieve the stress.

He takes a walk down Bubble Tea Street. There are N bubble tea shops on the street, numbered from 1 to N from the left. Some stores sell black bubble tea (i.e. with black tapioca balls), and some sell white. Nathan holds a record of these shops. The i-th shop sells black bubble tea if $c_i = 1$. Otherwise, it sells white and $c_i = 0$. Furthermore, there are exactly K shops which sell black bubble tea.

Nathan loves black bubble tea, so he usually buys all of the black bubble tea from the K shops. However, today he is too exhausted to resist the temptation of bubble tea. Therefore, for any L consecutive shops, Nathan will buy at least one cup of drink.

Nathan quickly realizes that under this constraint, he might buy some white bubble tea, which he hates. However, Nathan really wants to buy all K black bubble tea as he is totally devastated. To remedy this, Nathan asks his friend, Nahtan, to secretly swap the colors of bubble tea sold by some pairs of shops on the Bubble Tea Street. Nahtan agrees, but it is almost time for him to maintain the workstation. As a consequence, Nahtan can only swap at most Q pairs of shops.

They are running out of time. Given T different possible scenarios, can you help Nathan and Nahtan determine whether or not Nathan can buy all black bubble tea while avoiding the disgusting white bubble tea?

Input Format

T
testcase_1
testcase_2
$\vdots$
testcase_T

- T is the number of scenarios.
- testcase_i means the i-th scenario.
- Every scenario is independent from each other.

The input format for each scenario is as follows:

$N\ K\ L\ Q$
$c_1\ c_2\ \cdots\ c_N$

- N is the number of shops.

- K is the number of shops which sell black bubble tea.
- The meaning of L is described in the statement.
- Q represents the maximum amount of swaps Nahtan can perform.
- c_i is the color of bubble tea sold by the i-th shop.

Output Format

ans_1
ans_2
$\vdots$
ans_T

- ans_i is the answer to the i-th scenario, `"Yes"` if possible, `"No"` if not (without quotation marks).

Constraints

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq N \leq 2 \times 10^5$
- Let $\sum N$ be the sum of N over all scenarios in a testcase, $\sum N \leq 2 \times 10^5$.
- $1 \leq K, L \leq N$
- $0 \leq Q \leq 200$
- $c_i \in \{0, 1\}$
- Exactly K of the c_i equal to 1.
- All input are integers.

Sample Input

Sample Input	Sample Output
3 6 3 2 0 0 0 1 0 1 1 6 3 2 1 0 0 1 0 1 1 6 2 2 8 0 0 1 0 0 1	No Yes No

Scoring

In this problem, there are four subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	9	$\sum N \leq 500$
2	27	$K \leq 200$
3	29	$L \leq 20, Q \leq 20$
4	35	No additional constraints



This page is intentionally left blank.

C. Extra Sign-up Application (course)

Problem Statement

Professor Nya decided to give a course on *Algorithm Design and Analysis*, and the quota was instantly filled! Professor Nya hoped that more students can enroll in her course, so she opened N extra sign-up groups. In the end, the i-th group consists of a_i students.

As there are too many students who want to sign up for this course, the professor asks each group to play a game to decide their priorities:

- First, the a_i students form a circle and are numbered clockwise from 1 to a_i.
- The student with number 1 starts by counting off 1, and then the next student clockwise counts 2, and so on.
- Whenever someone says a number that is the sum of two non-negative powers of 2, he is eliminated and leaves the circle.
- The game continues until there's one student left.

Professor Nya is still pondering over how many students to accept from each group. As one of her TA, please tell her the elimination order of each group.

Input Format

$$
\begin{array}{l}
N \\
a_1 \\
a_2 \\
\vdots \\
a_N
\end{array}
$$

- N is the number of groups.
- a_i is the number of students in the i-th group.

Output Format

$$
\begin{array}{l}
b_{1,1}\ b_{1,2}\ \cdots\ b_{1,a_1-1} \\
b_{2,1}\ b_{2,2}\ \cdots\ b_{2,a_2-1} \\
\vdots \\
b_{N,1}\ b_{N,2}\ \cdots\ b_{N,a_N-1}
\end{array}
$$

- $b_{i,j}$ is the j-th person eliminated in the i-th group.

Constraints

- $1 \leq N \leq 10\ 000$
- $2 \leq a_i \leq 2 \times 10^5$
- $\sum_{i=1}^{N} a_i \leq 2 \times 10^5$

Sample Input

Sample Input	Sample Output
6 2 3 4 5 8 15	2 2 3 2 3 4 2 3 4 5 2 3 4 5 6 8 1 2 3 4 5 6 8 9 10 12 1 7 11 14 13

Scoring

In this problem, there are four subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	10	$a_i \leq 30, \sum_{i=1}^{N} a_i \leq 500$
2	20	$a_i \leq 120, \sum_{i=1}^{N} a_i \leq 8000$
3	30	$a_i \leq 1800, \sum_{i=1}^{N} a_i \leq 30\ 000$
4	40	No additional constraints

D. Farmland Management (farming)

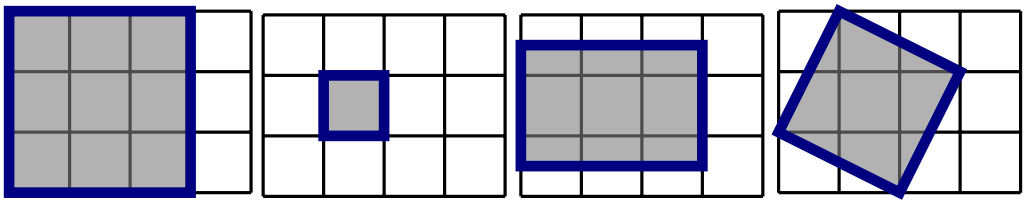
Problem Statement

We all remember that two years ago, Casper, the Great DongMeng Landlord, owned a lot of farmland in DongMeng, and he used pipes to connect the irrigation equipments.

Today, Casper has finally bought all the remaining land in DongMeng. With the entire DongMeng area under his administration, he now owns a large rectangular farmland. This is beneficial in all aspects, including simpler irrigation, more flexibility in space management, better utility ratio, and easier automation. He's now finally ready to start his grand farming project.

Casper partitions the land into a square grid of N rows and M columns, and labelled the square on the i-th row and the j column with coordinate (i, j). To manage the land, he decides to build an operating center on this rectangular land. The operating center will be rectangular, and it will use up all the space of some squares and none in the others.

For example, consider the following partition with $N = 3, M = 4$. The two plans on the left are valid, while the two on the right are not, because some suqares are partially covered.



Unfortunately, Casper didn't notice some of the squares are only for farming uses, and Casper will be taxed if he wants to build there. He doesn't have much money left, so his operating center can cover at most K such squares.

As the leading software engineer in Casper's team, he asks you to calculate the maximum possible size of a operating center, and count how many different operating centers can be built for each sizes.

Two plans for operating centers are considered different if they cover different set of squares.

Input Format

$N\ \ M\ \ K$
G_1
G_2
$\vdots$
G_N

- N, M are the number of rows and columns of Casper's farmland.
- The meaning of K is described in the statement.
- G_i is a string of length M, where $G_{i,j}$ is `1` if the sqaure with coordinate (i, j) is only for farming uses, and `0` otherwise.

Output Format

A
$C_1\ \ C_2\ \ C_3\ \ \cdots\ \ C_{NM}$

- A is the maximum possible size of the operating center. If there is no possible plan, A is defined to be 0.
- C_i is the number of different operating centers occupying i squares. This number doesn't have to be exact; see 'Scoring' section for details.

Constraints

- $1 \leq N, M \leq 2000$
- $0 \leq K \leq 5$
- G_i consists of only `01`.

Sample Input

Sample Input	Sample Output
3 4 1 0001 1000 0001	9 12 17 9 9 0 6 0 0 1 0 0 0
5 3 2 001 111 000 101 110	6 15 22 13 10 1 3 0 0 0 0 0 0 0 0 0

Scoring

For a testcase, if your program correctly calculates the maximum possible size of the operating center but fails to count ths numbers of different operating centers, you will get 40% of the score for that testcase. More precisely, for a testcase, your score S is calculated as follows:

- $S = 0.0$ if the format isn't correct.
- $S = 1.0$ if A and all C_i are correct.
- $S = 0.4$ if A is correct and all C_i satisfy $0 \leq C_i \leq 10^9$.
- $S = 0.0$ otherwise.

Note that in order to get 40% score, you might need to output $N \times M$ zeros in the second line.

There are three subtasks for the problem, with the following constraints. Each subtask contains one or more testcases. Your final score for a subtask is the minimum score across all testcases in that subtask multiplied by the points for that subtask.

Subtask	Points	Additional Constraints
1	5	$N, M \leq 100$
2	15	$N, M \leq 500$
3	80	No additional constraints



This page is intentionally left blank.

E. Nibbler Frieren (frieren)

Problem Statement

Frieren is so obsessed with magic that she often skips meals. Worried about her malnourished self, Mama Fern decides to make her more food. Mama Fern has already made N dishes and the i-th of them weighs a_i. Knowing that making any more dishes will just scare Frieren away, she decides that magic is the only way to make Frieren eat more.

Mama Fern just learned "The magic that makes food heavier", but it's not perfect. While increasing the weight of one dish, it comes at the cost of the weight of another dish. More specifically, applying "The magic that makes food heavier" once can double the weight of a dish that **hasn't been affected by the magic**, but Fern also has to choose another dish that **hasn't been affected by the magic** and divide its weight by 2, rounding down to the nearest integer. Note that this means if there's only one dish left that hasn't been affected by the magic, Fern can't use the magic at all.

Fern has enough mana to apply the magic as many times as she wants. She wonders what is the maximum amount of food she can make Frieren eat? Please help her solve this problem.

Input Format

N
$a_1\ a_2\ \ldots\ a_N$

- N is the number of dishes.
- a_i is the initial weight of the i-th dish.

Output Format

ans

- ans is the maximum sum of weights that can be obtained after using the magic.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq a_i \leq 10^9$

Sample Input

Sample Input	Sample Output
5 1 5 2 4 3	22

Scoring

In this problem, there are three subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	5	$N \leq 10$
2	9	$N \leq 2000$
3	86	No additional constraints

F. Level Design (level)

Problem Statement

Pixelcat is a new game designer. He's now troubled by his final project, a racing game.

In this game, some levels have more challenging levels locked behind them. In other words, the relationships between levels can be modeled as a rooted forest, where a level is locked behind its parent, if any.

Storing the relationship between levels can use up too much space. As an expert in designing algorithms, Pixelcat decides to store the relationships using a regular bracket sequence. Recall that a bracket sequence is a string that only consists of `(` and `)`, and a regular bracket sequence is recursively defined as follows:

- An empty string is regular.
- If S is regular, then `(`S`)` is regular.
- If S_1, S_2 are regular, then their concatenation is regular.

A critical error has happened and the level structure data has got corrupted. As the deadline is close, Pixelcat doesn't have enough time to restore the data manually. Fortunately, from the error log, he knows how many characters are altered in this incident. Please help Pixelcat recover one possible level structure, or tell him that the original data was wrong.

Input Format

N K
S

- N is the length of the string to restore.
- K is the number of altered characters.
- S is a string of length N, which is the corrupted data.

Output Format

If it is possible to restore the level structure, output: is

S'

- S' is the restored level data. S' has to be regular, and there are exactly K indices where S and S' differ. If there are many possible restoration, any one of them is considered correct.

Otherwise, output:

```
-1
```

Constraints

- $1 \leq N \leq 2 \times 10^6$
- $0 \leq K \leq N$
- S consists of only `(` and `)`.

Sample Input

Sample Input	Sample Output
4 1 ()((	()()
10 6))))))(((((	()(()()())
2 1)(	-1
4 4 (())	-1
4 4)()(	()()

Scoring

In this problem, there are three subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	12	$N \leq 200$
2	36	$N \leq 5000$
3	52	No additional constraints

G. Lunch Time (lunch)

Problem Statement

It's National Giraffe University's lunchtime, and N people are about to have lunch. To save time while not feeling lonely, they decide to split all of them into groups of two. You can assume that nobody will be left alone, that is, N is even. The procedure for having lunch is as follows.

- Riding bicycles to the food street nearby.
- Enjoying their meal.
- Gathering in the cafeteria and waiting until everyone arrives.
- Preparing for the upcoming exam together.

However, it takes everyone different time units to ride the bicycle and eat. Specifically, it takes the i-th person a_i units of time to ride the bicycle to the food street and b_i units of time to eat. In addition, two people in the same group will wait for each other for both riding the bicycle and eating. In other words, the person who rides the bicycle faster will wait for his partner to arrive before having lunch together, and the person who eats faster will wait for his partner to finish his meal. You don't have to consider the time they spend choosing restaurants, waiting for dishes, and walking to the cafeteria on the food street.

In other words, if the i-th person and the j-th person are in the same group, then their total time spent would be $\max(a_i, a_j) + \max(b_i, b_j)$.

Now, the N people want to split into groups faster so that they can have their lunch earlier and save more time to prepare for the exam. Could you help them calculate the minimum time to finish their lunch?

Input Format

N
$a_1\ a_2\ \ldots\ a_N$
$b_1\ b_2\ \ldots\ b_N$

- N is the number of people that are about to have lunch.
- a_i is the time it takes for the i-th person to ride a bicycle to the food street.
- b_i is the time it takes for the i-th person to eat.

Output Format

ans

- ans is the minimum time for all people to finish their lunch.

Constraints

- $2 \leq N \leq 10^5$
- $1 \leq a_i, b_i \leq 10^8$
- N is even.

Sample Input

Sample Input	Sample Output
6 3 2 1 4 1 3 1 3 2 1 4 2	6
10 3 9 10 39 77 98 73 87 28 23 42 97 87 74 28 34 8 6 65 55	132
10 46 99 6 99 38 43 34 36 15 94 53 47 54 32 53 2 2 84 81 65	159

Scoring

In this problem, there are three subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	15	$N \leq 20$
2	16	$N \leq 2000$
3	69	No additional constraints

H. Ore Excavation (ore)

Problem Statement

Pixel Mountain is one of the great wonders of the world. Legend has it that if someone finds the hidden treasure buried under the mountain, they can obtain the power of heaping trees.

After investigation, you found a possible site for the hidden treasure.

As its name implies, Pixel Mountain can be divided into mutliple cubes of identical size. The site you found contains N vertical mineshaft from left to right and adjacent mineshafts are connected by tunnels. You suspect that the treasure is hidden at some height, so let's call that height 0. Let (x, y) denote the position that is y cubes above 0 in the x-th mineshaft.

For mineshaft i, there are a_i cubes of ore sand above 0. You can only stand on top of those ore sand. You plan to start from the top of mineshaft 1 (i.e. $(1, a_1)$) and go to the bottom. You have brought a strong shovel to mine, and a piece of equipment to block damage. The equipment can block at most K damage, which is the most damage you can sustain.

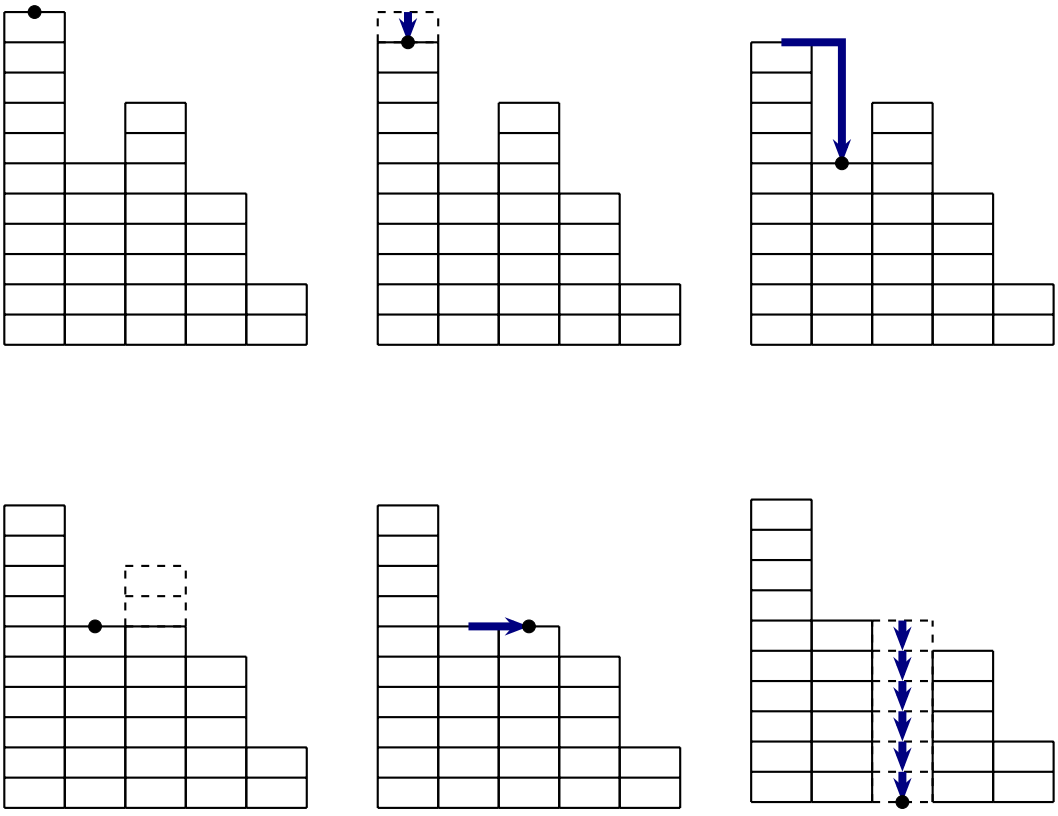
In each time unit, you can do one of the following:

1. Move one mineshaft to the left or right.
   - Suppose you're currently at mineshaft x, you can move to mineshaft $x-1$ or $x+1$, but you can't move left in the first mineshaft, nor move right in the N-th mineshaft.
   - The height of ore in the destination has to be lower than or equal to the height of ore in the current mineshaft.
   - Let the height of ore in the current mineshaft be y and in the destination be y'. You will drop to height y' in the destination, and receive $(y-y')^2$ damage.
2. Excavate some ore sand.
   - Suppose you're standing on (x, y), you can mine all ore sand **above** $(x-1, y)$, $(x+1, y)$ or $(x, y-1)$.
   - If you choose to mine the piece of ore sand at $(x, y-1)$, you will drop down and recieve 1 damage.

For example, when $N = 5, K = 30, a = \{11, 6, 8, 5, 2\}$, if you want to go from $(1, 11)$ to $(3, 0)$, this is one of the fastest way possible:

- Mine all ore sand above $(1, 10)$, and $(1, 11)$ disappears. You will fall down and recieve 1 damage.
- Move to mineshaft number 2, You will drop down and recieve 16 damage.
- Mine all ore sand above $(3, 6)$, and $(3, 7)$, $(3, 8)$ disappear in the same time unit.
- Move to mineshaft number 3, You won't recieve any damage in this step.
- Sequentially mine all ore sand above $(3, 5)$, $(3, 4)$, $\ldots$ until you reach $(3, 0)$, You will drop down and recieve 6 damage in total and it takes 6 unit of time.

Total time spent is 10 unit.



Pixel Mountain is a dangerous place and you want to reach the treasure as fast as possible. However, you don't know where exactly is the treasure, so you want to know, for $i = 1, 2, \ldots, N$, what is the shortest time to reach $(i, 0)$?

Input Format

$N\ K$
$a_1\ a_2\ \cdots\ a_N$

- N is the number of mineshafts.
- K is the most damage your equipment can sustain.
- a_i is the height of ore sand of the i-th mineshaft.

Output Format

$t_1\ t_2\ \cdots\ t_N$

- t_i is the minimum possible time to reach $(i, 0)$.

Constraints

- $1 \leq N \leq 10^6$
- $0 \leq a_i \leq 10^9$
- $a_1 \leq K \leq 10^{18}$

Note that it's always possible to reach the bottom of any mineshaft in finite time under these constraints.

Sample Input

Sample Input	Sample Output
5 11 11 6 8 5 2	11 11 13 13 13
5 30 11 6 8 5 2	10 8 10 10 8
6 100 36 30 24 18 12 6	30 28 26 24 22 21
2 998244853993244853 1000000000 0	877962 877960

Scoring

In this problem, there are five subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	6	$K = a_1$
2	13	$N \leq 1000, a_1 \leq 1000$
3	22	$N \leq 1000$
4	47	$N \leq 10^5$
5	12	No additional constraints

I. Transportation Vouchers (voucher)

Problem Statement

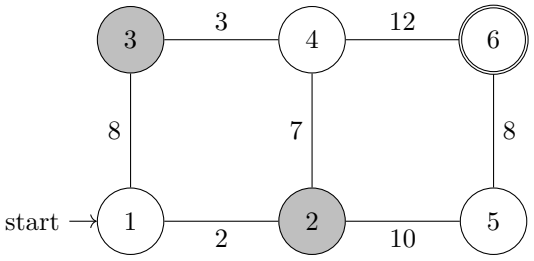
The NHSPC Nation is known for its cutting-edge technology. As the saying goes, technology comes from human nature, but here, people have been constantly working. With these technological advancements, the citizens have become even more hard-working. Ultimately, many have lost their work-life balance and suffered from this lifestyle. People have been physically and mentally exhausted for a long time, and the rate of various illnesses reaches new highs every year.

Fortunately, the new mayor is determined to rescue the nation. Therefore, he launched several events with the hope of arousing civilians' awareness of living well. One of the events is about the transportation vouchers. There are N cities in the NHSPC Nation, numbered from 1 to N. Additionally, M ultrasonic-bullet trains run through the city, i-th of which runs directly between u_i and v_i, costing c_i to travel from one side to the other. During the event, there are vouchers available in city A, B and they work as follows:

- One person can get a voucher from the same city only once during the event.
- After redeeming a voucher, one can select exactly one of the M trains, and it will be free during the event (which is setting one of c_i to 0).
- The effect of vouchers will not affect the cost spent on any train before it is redeemed.

Note that it means after claiming all the vouchers, one can set the cost of at most two trains to zero.

Consider the following example:



The two cities with vouchers are city 2 and 3. If one wants to travel from city 1 to city 6, there are several possibilities:

- Start from city 1, go to city 2, set the cost of the train between city 2 and 5 to zero, and go to city 5, 6. The total cost is 10.
- Start from city 1, go to city 2, set the cost of the train between city 4 and 6 to zero, and go to city 4, 6. The total cost is 9.
- Start from city 1, go to city 2, set the cost of the train between city 1 and 3 to zero, and go to city 1, 3, set the cost of the train between city 4 and 6 to zero, go to city 4, 6. The total cost is 7.

The minimum cost among all possibilities is 7.

If the cities with vouchers are changed to 3, 6, then the minimum cost travel plan is starting from city 1, traveling to city 3, setting the cost of the train between 4 and 6 to zero, then traveling to city 4, 6. The total cost is 11.

You think it is the right time to move to another city and get a new job! Therefore you need to know if you are currently in city S and plan to move to city T, what is the minimum cost of traveling from S to T by ultrasonic-bullet trains? As the last work during this event, please write a program to calculate the minimum cost.

Input Format

$N\ M\ S\ T\ A\ B$
$u_1\ v_1\ c_1$
$u_2\ v_2\ c_2$
$\vdots$
$u_M\ v_M\ c_M$

- N is the number of cities.
- M is the number of ultrasonic-bullet trains.
- S is the starting city and T is the target city.
- A, B are the cities giving out the vouchers.
- u_i, v_i, c_i denotes the two endpoints and cost of i-th train, respectively.

Output Format

d

- d represents the minimum transportation cost to get from S to T.

Constraints

- $2 \leq N \leq 3 \times 10^5$
- $1 \leq M \leq 5 \times 10^5$
- $1 \leq A, B, S, T \leq N$
- $A \neq B$
- $1 \leq u_i < v_i \leq N$
- $(u_i, v_i) \neq (u_j, v_j) \quad (i \neq j)$
- $1 \leq c_i \leq 10^9$
- Every city can be reached from other cities along a series of trains.
- All inputs are integers.

Examples

Sample Input	Sample Output
6 7 1 6 2 3 1 2 2 1 3 8 2 4 7 2 5 10 3 4 3 4 6 12 5 6 8	7
6 7 1 6 6 3 1 2 2 1 3 8 2 4 7 2 5 10 3 4 3 4 6 12 5 6 8	11

Scoring

In this problem, there are five subtasks with the following constraints. Each subtask contains one or more testcases. You need to pass all testcases in a subtask to receive the points for that subtask.

Subtask	Points	Additional Constraints
1	21	$A = T$
2	9	$M \leq 100$
3	17	$M \leq 2000$
4	35	$N \leq 80\ 000, M \leq 10^5$
5	18	No additional constraints