15th JOI Selection Final Selection Tasks (2016)

Authors: JCIOI (Japanese Committee for the IOI)

Translators: Jasmine Chua, France-ioi

License: Creative Commons Attributions-ShareAlike 4.0 International License (CC BY-SA 4.0)

**1. Oranges**

Do you know the company Juicy Orange Industry? Their business is to cultivate delicious oranges and ship them to customers. They are also known as JOI for short.

JOI Company is packing N oranges into crates for shipping. The oranges are lined up in a row on a conveyor belt at their plant, and the oranges are labeled from 1 to N. The oranges are of different sizes. Orange i, where 1 ≦ i ≦ N, has a size of Ai.

The oranges are to be packed into several crates in order, starting from Orange 1. Oranges must be placed into the same crate consecutively, meaning that there is no way to add oranges to a previous crate once they have started packing the next crate.

Each crate can hold a maximum of M oranges. The cost of packing the oranges in each crate is calculated by the formula K + s × (a − b), where a is the size of the largest orange, b is the size of the smallest orange, s is the number of oranges in that crate, and K is a fixed value across all crates.

JOI Company has prepared several crates to pack all the oranges, and they want to minimize the cost of packing as much as possible.

**QUESTION**

Given information regarding the oranges on the conveyor belt, the maximum number of oranges that can be packed into a crate, and the fixed price for packing, create a program to find out the minimum cost to pack all the oranges.

**INPUT**

In the first row, three integers are written with a space between them.

* N, representing the number of oranges.
* M, representing the maximum number of oranges that can be packed into a crate.
* K, representing the fixed price for packing.

For the next N rows, in the ith row, where 1 ≦ i ≦ N, one integer is written.

* Ai, representing the size of Orange i.

**OUTPUT**

Output one row, with the minimum cost to pack all the oranges.

**CONSTRAINTS**

All input data must fulfil the following conditions:

* 1 ≦ N ≦ 20 000
* 1 ≦ M ≦ 1 000
* 0 ≦ K ≦ 1 000 000 000
* 1 ≦ Ai ≦ 1 000 000 000, where 1 ≦ i ≦ N
* M ≦ N

**SUBTASKS**

**Subtask 1 [20 points]**

Fulfil the following constraint:

* N ≦ 20

**Subtask 2 [50 points]**

Fulfil the following constraint:

* N ≦ 2 000
* M ≦ 100

**Subtask 3 [30 points]**

No additional constraints.

**INPUT/OUTPUT EXAMPLES**

|  |  |
| --- | --- |
| INPUT 1 | OUTPUT 1 |
| 6 3 6  1  2  3  1  2  1 | 21 |

Oranges 1 to 3 will be packed into the first crate, and Oranges 4 to 6 will be packed into the second crate. The total cost of packing is (6+3×(3−1))+(6+3×(2−1)) = 21.

Since 21 is the lowest possible cost, the output is 21.

|  |  |
| --- | --- |
| INPUT 2 | OUTPUT 2 |
| 16 4 12  3  10  13  10  19  9  12  16  11  2  19  9  13  2  13  19 | 164 |

There will be 11 boxes with 1, 3, 1, 1, 3, 1, 1, 2, 1, 1, 1 oranges in each respectively. This will result in the lowest possible cost.

|  |  |
| --- | --- |
| INPUT 3 | OUTPUT 3 |
| 16 6 14  19  7  2  15  17  7  14  12  3  14  5  10  17  20  19  12 | 177 |

|  |  |
| --- | --- |
| INPUT 4 | OUTPUT 4 |
| 10 1 1000000000  1  1  1  1  1  1  1  1  1  1 | 10000000000 |

Note that the answer may not fit into the 32-bit integer range.

**2. Collecting Stamps 2**

There are N shops lined up along JOI shopping district, labeled 1, 2, …, N starting from the entrance of the district. The shopping district is a one-way road, and shoppers can only move from the entrance towards the exit.

As part of a revitalization project, JOI shopping district is holding a stamp rally event. Each shop in the district will prepare a stamp with one of the following letters: J, O, I. Any shopper who makes a purchase at the shop can get a stamp on their stamp card. Shoppers who wish to take part in this event will be given a stamp card with 3 boxes at the start of the shopping district. They will have to shop at 3 different shops to get the card stamped in order, from the leftmost box to the rightmost box. The card will be collected at the exit of the shopping district. Only shoppers whose cards have the stamps lined up to form the word 'J,O,I' will get a voucher.

All N shops have already prepared their stamps, but 1 new shop will be opened in the shopping district. Currently, the shop is deciding its location, and which stamp to prepare. The new shop can choose to open between Shops i and (i + 1), where 1 ≦ i ≦ N − 1, between the entrance and Shop 1, or between Shop N and the exit. In addition, the new shop can choose to have any 1 of the 3 stamps, J, O, or I.

Helping shoppers to get free vouchers will help to revitalize the shopping district. Thus, the new shop wants to choose the best location and stamp, so shoppers have maximum chance of winning.

**QUESTION**

Giving information regarding the stamps prepared by all shops in JOI shopping district, create a program to find out the maximum number of ways shoppers can visit shops to get their card stamped right to win a voucher, after determining the new shop's location and stamp.

**INPUT**

In the first row, integer N is written. This represents that there are N shops in JOI shopping district.

In the second row, a string of N characters, S, is written. The characters can be J, O, or I. The ith character from the left, where 1 ≦ i ≦ N, represents the stamp prepared by Shop i.

**OUTPUT**

Output the maximum number of ways shoppers can visit shops to get their card stamped right to win a voucher in one row.

Note that the output may not fit into a 32-bit integer.

**CONSTRAINTS**

All input data must fulfil the following constraint:

* 3 ≦ N ≦ 100 000

**SUBTASKS**

**Subtask 1 [30 points]**

Fulfil the following constraint:

* N ≦ 200

**Subtask 2 [20 points]**

Fulfil the following constraint:

* N ≦ 3 000

**Subtask 3 [50 points]**

No additional constraints.

**INPUT/OUTPUT EXAMPLES**

|  |  |
| --- | --- |
| INPUT 1 | OUTPUT 1 |
| 5  JOIOI | 6 |

In INPUT EXAMPLE 1, the new shop will open between Shops 1 and 2 and prepare a 'J' stamp. Thus, the sequence of stamps from the entrance becomes JJOIOI.

In this case, there are 6 ways to visit shops to win a voucher.

* Visit Shops 1, 3, and 4.
* Visit Shops 1, 3, and 6.
* Visit Shops 1, 5, and 6.
* Visit Shops 2, 3, and 4.
* Visit Shops 2, 3, and 6.
* Visit Shops 2, 5, and 6.

6 is the maximum number of ways to win a voucher.

|  |  |
| --- | --- |
| INPUT 2 | OUTPUT 2 |
| 7  JJJOIII | 18 |

|  |  |
| --- | --- |
| INPUT 3 | OUTPUT 3 |
| 4  OIIJ | 2 |

In INPUT EXAMPLE 3, the new shop will open between the entrance and Shop 1 and prepare a 'J' stamp. This will result in the maximum number of ways to win a voucher.

**3. Train Fare**

There are N cities in JOI Kingdom, labeled 1, 2, ..., N. City 1 is JOI Kingdom's capital city.

There is only one railway company in JOI Kingdom, which operates a total of M lines, labeled 1, 2, ..., M. Line i, where 1 ≦ i ≦ M, runs between Cities Ui and Vi. The train lines are the only way to travel between cities. Passengers can transfer between multiple lines to get to any city.

Right now, the fare to travel on any line is 1 yen. However, the railway company is facing financial trouble, and is planning an initiative to raise the fare of several train lines over Q years. At the beginning of the jth year from the start of the initiative, where 1 ≦ j ≦ Q, the fare for Line Rj will increase from 1 yen to 2 yen. After the fare raise, the fare of that line will remain capped at 2 yen.

The railway company also conducts annual surveys with the residents of each city. All residents are currently satisfied with the railways before the initiative, but some residents may become dissatisfied once the fare hike happens.

The company will conduct each year's survey after implementing the fare hike. In other words, jth year's survey, where 1 ≦ j ≦ Q, will be conducted after the fare hike for Lines R1, R2, ..., Rj. All other lines are unaffected at that point in time. jth year's survey, where 1 ≦ j ≦ Q, will be conducted with the residents of City k, where 2 ≦ k ≦ N. If a resident fulfils the following condition, then they are dissatisfied with the company:

* The minimum fare needed to travel from City k to capital City 1 now, is higher than the minimum fare needed to travel from City k to capital City 1 before the initiative.

The fare is calculated based on the number of lines used due to transfers. In addition, the residents in City 1 are satisfied with the railways. Note that the route of travel between 2 cities that incurs the minimum fare after the fare hike, may be different from the route that incurs the minimum fare before the fare hike.

Before starting the initiative, the company wants to calculate how many cities will have dissatisfied residents in the Q years to come.

**QUESTION**

Given information about JOI Kingdom and the initiative to raise train fares, create a program to find out how many cities will have dissatisfied residents.

**INPUT**

In the first row, three integers are written with a space between them.

* N represents the number of cities in JOI Kingdom.
* M represents the number of train lines.
* Q represents the number of years the initiative will take place over.

For the next M rows, in the ith row, where 1 ≦ i ≦ M, two integers are written with a space between them.

* Ui and Vi represent the 2 cities connected by Line i.

For the next Q rows, in the jth row, where 1 ≦ j ≦ Q, one integer is written.

* Rj represents the train line whose fare is increased in the jth year.

**OUTPUT**

The output has Q rows. In the jth row, where 1 ≦ j ≦ Q, output the number of cities with dissatisfied residents.

**CONSTRAINTS**

All input data must fulfil the following constraints:

* 2 ≦ N ≦ 100 000
* 1 ≦ Q ≦ M ≦ 200 000
* 1 ≦ Ui ≦ N, where 1 ≦ i ≦ M
* 1 ≦ Vi ≦ N, where 1 ≦ i ≦ M
* Ui ≠ Vi, where 1 ≦ i ≦ M
* 1 ≦ Rj ≦ M, where 1 ≦ j ≦ Q
* Rj ≠ Rk, where 1 ≦ j < k ≦ Q
* There is only 1 line that directly joins any 2 cities.
* It is possible to travel from any city to capital City 1.

**SUBTASKS**

**Subtask 1 [12 points]**

Fulfil the following constraint:

* N ≦ 100
* M ≦ 4 950
* Q ≦ 30

**Subtask 2 [14 points]**

Fulfil the following constraint:

* Q ≦ 30

**Subtask 3 [35 points]**

Fulfil the following constraint:

* There are ≤50 unique integers in the output.

**Subtask 4 [39 points]**

No additional constraints.

**INPUT/OUTPUT EXAMPLES**

|  |  |
| --- | --- |
| INPUT 1 | OUTPUT 1 |
| 5 6 5  1 2  1 3  4 2  3 2  2 5  5 3  5  2  4  1  3 | 0  2  2  4  4 |

In INPUT EXAMPLE 1, the fare for travel from any city to capital City 1 is shown below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Time | City 2 | City 3 | City 4 | City 5 |
| Initiative start | 1 | 1 | 2 | 2 |
| Year 1 | 1 | 1 | 2 | 2 |
| Year 2 | 1 | 2 | 2 | 3 |
| Year 3 | 1 | 2 | 2 | 3 |
| Year 4 | 2 | 2 | 3 | 3 |
| Year 5 | 2 | 2 | 4 | 3 |

For example, the survey held in Year 3 will show that the residents in Cities 3 and 5 will be dissatisfied, so the output in the 3rd row is 2.

|  |  |
| --- | --- |
| INPUT 2 | OUTPUT 2 |
| 4 6 6  1 2  1 3  1 4  2 3  2 4  3 4  1  4  2  5  3  6 | 1  1  2  2  3  3 |

|  |  |
| --- | --- |
| INPUT 3 | OUTPUT 3 |
| 2 1 1  1 2  1 | 1 |

**4. Territory**

You live in a city where many long roads stretching from North to South intersect with many long roads stretching from East to West. The distance between each road and the road running parallel next to it is 1 km.

There is one City Hall in your city, located at intersection (0,0). Each intersection is labeled (i, j), where i and j are integers. Intersection (i, j) is i km to the east of intersection (0,0) (or −i km if to the west), and j km from the north of intersection (0,0) (or −j km if to the south).

The staff at City Hall keep a pet dog named JOI. JOI the dog will go on walks for K days. The walks follow the rules below:

* On the morning of the first day, JOI is at intersection (0,0). JOI leaves its mark at intersection (0,0). None of the other intersections has been marked by JOI.
* JOI will take a walk each afternoon for K days. Each walk consists of N events. An event includes JOI moving from one intersection to an adjacent one and leaving its mark there.
* After its walk for the day, JOI will sleep at the intersection it is currently at, until the next day.

The staff at City Hall are discussing about the territory JOI can mark throughout its walks over K days. JOI's territory is defined as the area bounded by 4 intersections (a,b), (a + 1,b), (a + 1,b + 1), and (a,b + 1), all of which JOI has left its mark at at least once.

You are tasked with creating a program to find out the area of JOI's territory, given JOI's walking plan.

The roads in the city are extremely long and numerous, so assume that JOI will not exit the boundaries of the city during its walks.

**QUESTION**

Given JOI's walking plan, create a program to find out the area of JOI's territory.

**INPUT**

In the first row, two integers are written with a space between them.

* N, representing the number of events that happen each day.
* K, representing the number of days of JOI's walking schedule.

In the second row, a string of N characters, S, is written. Cp, which is the pth character from the left, where 1 ≦ p ≦ N, can be either of the following:

* E, representing that event p is a move to the adjacent intersection on the east.
* N, representing that event p is a move to the adjacent intersection on the north.
* W, representing that event p is a move to the adjacent intersection on the west.
* S, representing that event p is a move to the adjacent intersection on the south.

For intersection (i, j), the intersections to its east, north, west, and south are (i + 1, j), (i, j + 1), (i − 1, j), and (i, j − 1) respectively.

**OUTPUT**

Output the area of JOI's territory in one row.

**CONSTRAINTS**

All input data must fulfil the following constraints:

* 1 ≦ N ≦ 100 000
* 1 ≦ K ≦ 1 000 000 000

**SUBTASKS**

**Subtask 1 [5 points]**

Fulfil the following constraints:

* N ≦ 50
* K = 1

**Subtask 2 [10 points]**

Fulfil the following constraint:

* K = 1

**Subtask 3 [23 points]**

Fulfil the following constraint:

* N ≦ 50

**Subtask 4 [62 points]**

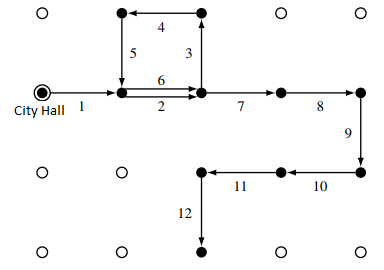
No additional constraints.

**INPUT/OUTPUT EXAMPLES**

|  |  |
| --- | --- |
| INPUT 1 | OUTPUT 1 |
| 12 1  EENWSEEESWWS | 3 |

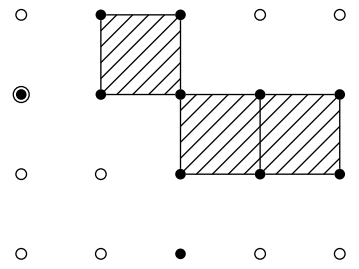
In INPUT EXAMPLE 1, the walking plan consists of 1 day. JOI's route is shown in the diagram below.

The black dots represent the intersections where JOI left its mark, and the white dots represent intersections JOI has not marked. The black dot with the ring represents City Hall, and the numbers represent the events in sequence.



JOI's route

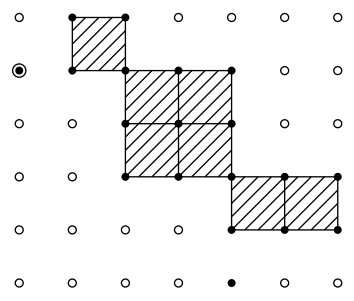
JOI's territory is marked out by the 3 shaded areas shown below.



JOI's territory in INPUT EXAMPLE 1

|  |  |
| --- | --- |
| INPUT 2 | OUTPUT 2 |
| 12 2  EENWSEEESWWS | 7 |

In INPUT EXAMPLE 2, the walking plan consists of 2 days. JOI's route is the same as in INPUT EXAMPLE 1. JOI's territory after 2 days is marked out by the 7 shaded areas shown below.



JOI's territory in INPUT EXAMPLE 2

Note that INPUT EXAMPLE 2 does not fulfil the constraints in Subtasks 1 and 2.

|  |  |
| --- | --- |
| INPUT 3 | OUTPUT 3 |
| 7 1  ENNWNNE | 0 |

In INPUT EXAMPLE 3, JOI does not have any territory.

|  |  |
| --- | --- |
| INPUT 4 | OUTPUT 4 |
| 16 5  WSESSSWWWEEENNNW | 21 |

Note that INPUT EXAMPLE 4 does not fulfil the constraints in Subtasks 1 and 2.

**5. Geologic Fault**

A long time ago, there was an ancient civilization named IOI. However, the civilization was wiped out by a volcanic eruption. IOI civilization was located along a straight river, which was a flat piece of land when the eruption occurred. The location of IOI civilization's ruins can be mapped on an x-axis following the river, with the y-axis representing height. In other words, y = 0 represents ground level, y > 0 represents a height above ground, and y < 0 represents a height underground. The layers of earth from a years ago since the eruption, where a ≧ 0, is located at y = −a.

After the eruption, Q geologic events happened at the ruins. The ith geologic event, where 1 ≦ i ≦ Q, will create a fault at position Xi, with a direction of Di, and generate movement with a degree of Li. Di can be 1 or 2. In the ith geologic event, the following steps occur:

* The earth will move as follows:
  + When Di = 1, a fault will form at position (Xi, 0) and stretch diagonally down with a gradient of 1. All regions above the fault will move up along the fault by a height of Li. In other words, point (x, y) above the fault will move to (x + Li, y + Li).
  + When Di = 2, a fault will form at position (Xi, 0) and stretch diagonally down with a gradient of -1. All regions above the fault will move up along the fault by a height of Li. In other words, point (x, y) above the fault will move to (x − Li, y + Li).
* Immediately after this, all regions where y > 0 are eroded away by wind.

Many years pass, and in the present day, archeologist Professor JOI wants to excavate the ruins of IOI civilization. He wants to know what are the layers of earth that are present on the surface now. He has information on all geologic events that have occurred. Your job is to help Professor JOI find out the layers of earth that are present on the surface between points (i−1, 0) and (i, 0), where 1 ≦ i ≦ N.

**QUESTION**

Given information about the geologic events, find out the layers of earth that are present on the surface of all points i, where 1 ≦ i ≦ N.

**INPUT**

In the first row, two integers are written with a space between them.

* N, representing the number of points required by the answer.
* Q, representing the number of geologic events.

For the next Q rows, in the ith row, where 1 ≦ i ≦ Q, three integers are written with a space between them.

* Xi, representing the position of the fault.
* Di, representing the direction of the fault.
* Li, representing the degree of the movement.

**OUTPUT**

The output has N rows. Output the layers of earth that are present on the surface of all points i, where 1 ≦ i ≦ N.

**CONSTRAINTS**

All input data must fulfil the following constraints:

* 1 ≦ N ≦ 200 000
* 1 ≦ Q ≦ 200 000
* −1 000 000 000 ≦ Xi ≦ 1 000 000 000, where 1 ≦ i ≦ Q
* 1 ≦ Di ≦ 2, where 1 ≦ i ≦ Q
* 1 ≦ Li ≦ 1 000 000 000, where 1 ≦ i ≦ Q

**SUBTASKS**

**Subtask 1 [18 points]**

Fulfil the following constraints:

* N ≦ 100
* Q ≦ 100
* −100 ≦ Xi ≦ 100, where 1 ≦ i ≦ Q
* Li = 1, where 1 ≦ i ≦ Q

**Subtask 2 [16 points]**

Fulfil the following constraints:

* N ≦ 3 000
* Q ≦ 3 000

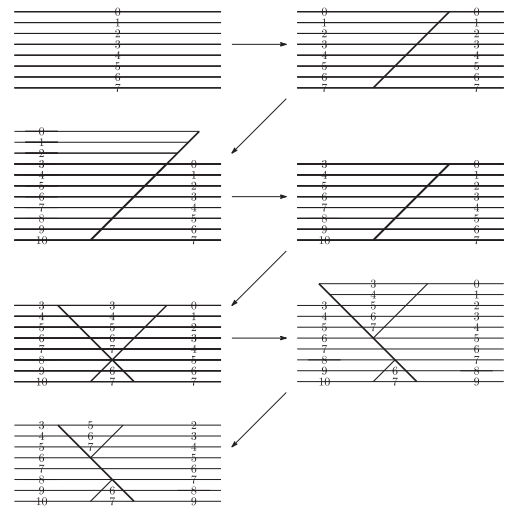
**Subtask 3 [66 points]**

No additional constraints.

**INPUT/OUTPUT EXAMPLES**

|  |  |
| --- | --- |
| INPUT 1 | OUTPUT 1 |
| 10 2  12 1 3  2 2 2 | 3  3  5  5  5  5  5  5  2  2 |

The diagrams below correspond to INPUT EXAMPLE 1.



|  |  |
| --- | --- |
| INPUT 2 | OUTPUT 2 |
| 10 6  14 1 1  17 1 1  -6 2 1  3 2 1  4 1 1  0 2 1 | 5  5  4  5  5  5  5  5  4  4 |

|  |  |
| --- | --- |
| INPUT 3 | OUTPUT 3 |
| 15 10  28 1 7  -24 2 1  1 1 1  8 1 1  6 2 1  20 1 3  12 2 2  -10 1 3  7 2 1  5 1 2 | 15  14  14  14  14  12  12  12  12  12  12  12  15  15  12 |