**JOI 2017-2018 Qualification contest**

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)

**Pencils (Allocated marks: 100)**

**QUESTION**

JOI went to a stationery store nearby to buy N pencils.

The stationery store sells pencils in sets containing a certain number. Set X has A pencils and costs B yen. Set Y has C pencils and costs D yen.

JOI chooses either Set X or Set Y and buys a few sets of that type. He is unable to buy sets of both types. What is the minimum amount of money he needs to buy ≥N pencils?

**CONDITIONS**

* 1 ≦ N ≦ 1000
* 1 ≦ A ≦ 1000
* 1 ≦ B ≦ 1000
* 1 ≦ C ≦ 1000
* 1 ≦ D ≦ 1000

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

N A B C D

**OUTPUT**

Output the minimum amount of money JOI needs to buy ≥N pencils.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

10 3 100 5 180

**OUTPUT 1**

360

JOI wants to buy 10 pencils. Set X has 3 pencils and costs 100 yen. Set Y has 5 pencils and costs 180 yen. If he chooses to buy Set X, he needs to buy 4 sets, which costs 400 yen. If he chooses to buy Set Y, he needs to buy 2 sets, which costs 360 yen. Therefore, comparing 400 and 360 yen, 360 yen is the minimum amount of money needed.

**INPUT 2**

6 2 200 3 300

**OUTPUT 2**

600

In this case, choosing either Set X or Set Y will cost 600 yen. Therefore, 600 yen is the minimum amount of money needed.

**Sugoroku (Allocated marks: 100)**

**QUESTION**

JOI finds a Sugoroku game set at his uncle's house. Sugoroku consists of N+2 squares arranged in a straight line. The first square is the start, and the (N+2)th square is the goal. Each of the other squares have a 0 or 1 written on it. For each square i, where (1≦i≦N), the (i+1)th square will have the number A\_i written on it.

In Sugoroku, players place their counters in the start square, then roll a die to determine the number of squares they will move, and this process repeats. However, if they come to stop on a square with 1 written on it, they lose. To clear the game, they need to stop on the goal square without losing or move past the goal square.

To play the game, JOI went to a toy store to buy a die. The toy store sells N+1 dice. The jth die, where (1≦j≦N+1), has j sides, with numbers 1,2,...,j written on each.

JOI will buy 1 die with the minimum amount of sides needed to clear the game. Which die should JOI buy?

**CONDITIONS**

* 1≦N≦100
* 0≦A\_i≦1 (1≦i≦N)

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

N

A\_1 A\_2 ... A\_N

**OUTPUT**

Output the amount of sides of the die JOI should buy.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

5

0 1 0 0 0

**OUTPUT 1**

2

The Sugoroku board has 7 squares, and the 3rd square has 1 written on it. If he uses a die with 2 sides and gets the following sequence: 1,2,1,1,1, he can clear the game. This is the minimum amount, so the output is 2.

**INPUT 2**

5

1 1 1 1 1

**OUTPUT 2**

6

The Sugoroku board has 7 squares, and all squares except start and goal have 1 written on them. In this case, he needs a die with 6 sides. This is the minimum amount, so the output is 6.

**INPUT 3**

7

0 0 1 0 1 1 0

**OUTPUT 3**

3

**Trunk Road (Allocated marks: 100)**

**QUESTION**

JOI City is segregated into a grid, with H straight roads running from West to East, and W straight roads running from North to South. The distance between consecutive roads is 1. In JOI City, out of the total number of H+W roads, 1 West-East road and 1 North-South road, for a total of 2 roads, were chosen to be trunk roads.

The intersection between the ith road from the North, where (1≦i≦H), and the jth road from the West, where (1≦j≦W), is known as the (i,j) intersection. The distance between the intersection (i,j) and the mth road from the North, where (1≦m≦H), is |i-m| meters. The distance between the intersection (i,j) and the nth road from the West, where (1≦n≦W), is |j-n| meters. In addition, there are A\_{i,j} number of people living near the (i,j) intersection.

When the 2 trunk roads were chosen, what is the minimum sum of the distance between their closest intersection and the nearest trunk road for all the people living in JOI city?

**CONDITIONS**

* 2 ≦ H ≦ 25
* 2 ≦ W ≦ 25
* 0 ≦ A\_{i,j} ≦ 100 (1 ≦ i ≦ H, 1 ≦ j ≦ W)

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

H W

A\_{1,1} A\_{1,2} ... A\_{1,W}

:

A\_{H,1} A\_{H,2} ... A\_{H,W}

**OUTPUT**

Output the minimum sum of the distance between their closest intersection and the nearest trunk road for all the people living in JOI city.

**TASKS**

**TASK 1 (10 marks)**

* A\_{i,j} = 1 (1 ≦ i ≦ H, 1 ≦ j ≦ W)

**TASK 2 (90 marks)**

* No additional conditions.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

3 5

1 1 1 1 1

1 1 1 1 1

1 1 1 1 1

**OUTPUT 1**

8

For example, the 2nd road from the North and the 1st road from the East should be chosen as trunk roads.

**INPUT 2**

5 5

1 2 3 1 5

1 22 11 44 3

1 33 41 53 2

4 92 35 23 1

4 2 6 3 5

**OUTPUT 2**

164

**Mizuyokan (Allocated marks: 100)**

**QUESTION**

Mizuyokan are a Japanese sweet made primarily from red bean paste placed into a mold and solidified using agar. JOI has 1 mizuyokan block in the shape of an oblong cuboid. He plans to have some mizuyokan as a snack.

The mizuyokan has N-1 notches in the longitudinal direction where it can be cut. The length of the mizuyokan is L\_1 + L\_2 + ... + L\_N. The ith cut, where (1 ≦ i ≦ N-1), is at a position L\_1 + L\_2 + ... + L\_i from the left.

The mizuyokan is too big to be eaten in one go, so JOI chooses more than 1 notch, and cuts the mizuyokan into smaller pieces following those notches. However, since the mizuyokan will look unappetizing if cut into pieces of differing sizes, he tried his best to minimize the size difference between the biggest and smallest pieces.

What is the minimum difference between the largest and smallest pieces?

**CONDITIONS**

* 2 ≦ N ≦ 50
* 1 ≦ L\_i ≦ 1000 (1 ≦ i ≦ N)

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

N

L\_1

L\_2

:

L\_N

**OUTPUT**

Output the minimum difference between the largest and smallest pieces in one row.

**TASKS**

**TASK 1 (10 marks)**

* N ≦ 15

**TASK 2 (27 marks)**

* L\_i ≦ 10 (1 ≦ i ≦ N)

**TASK 3 (63 marks)**

* No additional conditions.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

11

2

3

8

4

7

6

6

5

1

7

5

**OUTPUT 1**

2

In this example, if he makes cuts at the 4th and 7th notches, he can cut the mizuyokan into 3 pieces, with lengths 17, 19, and 18. The longest piece is 19 and the shortest piece is 17, so the difference is 2. This is the minimum amount, so the output is 2.

**INPUT 2**

2

1

10

**OUTPUT 2**

9

No matter the difference in sizes, he must make at least 2 cuts.

**INPUT 3**

5

5

5

5

5

5

**OUTPUT 3**

0

In this example, he could divide the mizuyokan equally into 5 pieces of the same size.

**Deforestation (Allocated marks: 100)**

**QUESTION**

There is a large forest in JOI Kingdom. The forest covers a rectangular area and is segregated into a grid, with H squares from North to South, and W squares from West to East. The region in the ith square from the North and jth square from the West has A\_{i,j} trees growing on it. However, there is a wood processing plant in the most north-western region, so no trees grow there. In other words, A\_{1,1}=0.

People can enter regions where no trees are growing. In addition, if the regions adjacent to their region in any direction has no trees, they can move into that region. They cannot leave the forest. JOI works as a civil servant in JOI Kingdom and does logging work, and he wants to allow people to travel freely between the most north-western region and most south-eastern region.

This is how JOI logs wood. He starts off in the most north-western region, at the wood processing plant. He can move to any adjacent region where no trees are growing in 1 minute. Once there, he can cut down 1 tree from an adjacent region where trees are growing in 1 minute. Every time he cuts down a tree, he needs to transport it back to the wood processing plant. While transporting it, his movement speed does not change. He also cannot cut down other trees while transporting a tree.

What is the minimum amount of time needed to cut down trees that satisfies the conditions above? Please include the time he needs to transport the last cut tree to the wood processing plant.

**CONDITIONS**

* 1 ≦ H ≦ 30
* 1 ≦ W ≦ 30
* (H, W) ≠ (1, 1)
* 0 ≦ A\_{i,j} ≦ 10000 (1 ≦ i ≦ H, 1 ≦ j ≦ W)
* A\_{1,1}=0

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

H W

A\_{1,1} ... A\_{1,W}

:

A\_{H,1} ... A\_{H,W}

**OUTPUT**

Output one line containing the minimum amount of time needed to cut down trees that satisfies the conditions.

**TASKS**

**TASK 1 (15 marks)**

* 1 ≦ H ≦ 5
* 1 ≦ W ≦ 5

**TASK 2 (28 marks)**

* A\_{i,j} ≦ A\_{i,j+1} (1 ≦ i ≦ H, 1 ≦ j ≦ W-1)
* A\_{i,j} ≦ A\_{i+1,j} (1 ≦ i ≦ H-1, 1 ≦ j ≦ W)

**TASK 3 (57 marks)**

* No additional conditions.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

2 3

0 1 2

3 4 5

**OUTPUT 1**

32

The region in the ith square from the North and jth square from the West is expressed as (i, j).

He will first cut down the tree in (1, 2). This will take 1 minute.

Next, he will cut down all the trees in (1, 3). He just needs to move 1 square East from (1,1), cut down a tree in (1, 3), and return to (1,1) by moving 1 square West, taking 3 minutes in total. Therefore, he needs 2 × 3 = 6 minutes to do this step.

Next, he will cut down all the trees in (2, 3). He needs to move 2 squares East from (1,1), cut down a tree in (2, 3), and return to (1,1) by moving 2 squares West, taking 5 minutes in total. Therefore, he needs 5 × 5 = 25 minutes to do this step.

In total, he needs 1 + 6 + 25 = 32 minutes. This is the minimum amount of time needed to cut down trees that satisfies the conditions, so the output is 32.

**INPUT 2**

2 5

0 5 0 0 0

0 0 0 9 1

**OUTPUT 2**

13

He just needs to cut down the tree in (2, 5).

**INPUT 3**

2 5

0 2 0 0 0

0 0 0 9 1

**OUTPUT 3**

11

First, he will cut down the trees in (1, 2), then cut down the tree in (2, 5).

**Lth's Kth Number (Allocated marks: 100)**

**QUESTION**

There are N cards arranged in a horizontal row. The ith card from the left, where (1 ≦ i ≦ N), has integer a\_i written on it.

JOI will use these cards to play the following game. He will choose a group of >K consecutive cards and do the following steps.

He will remove the chosen cards and rearrange them in ascending order of integers, with the card with the smallest integer on the left.

From this line, he will find the Kth card from the left, and write down its integer on a piece of paper.

He will return all cards to their original positions.

He will repeat these steps for all relevant lines of >K consecutive cards. In other words, for all (l,r) that satisfy the conditions 1 ≦ l ≦ r ≦ N and K ≦ r - l + 1, he will write down the Kth smallest integer out of a\_l, a\_{l+1}, ..., a\_r.

JOI will then rearrange all the integers he has written down in ascending order, with the smallest integer on the left. Out of this line, the Lth number from the left will be the points he gets from the game. How many points does he get?

**CONDITIONS**

* 1 ≦ N ≦ 200000
* 1 ≦ K ≦ N
* 1 ≦ a\_i ≦ N
* 1 ≦ L
* JOI writes down >L integers.

**INPUT/OUTPUT**

**INPUT**

Input can be derived from the standard input using the following format:

N K L

a\_1 a\_2 ... a\_N

**OUTPUT**

Output the points JOI gets in one row.

**TASKS**

**TASK 1 (6 marks)**

* N ≦ 100

**TASK 2 (33 marks)**

* N ≦ 4000

**TASK 3 (61 marks)**

* No additional constraints.

**INPUT/OUTPUT EXAMPLES**

**INPUT 1**

4 3 2

4 3 1 2

**OUTPUT 1**

3

There are three (l,r) that satisfy 1 ≦ l ≦ r ≦ N (= 4) and K (= 3) ≦ r - l + 1: (1,3), (1,4), and (2,4).

For these three (l,r), the 3rd smallest integer out of a\_l, a\_{l+1}, ..., a\_r is 4, 3, and 3 respectively.

In this case, the L (= 2) th smallest integer is 3, so JOI gets 3 points. Note that even when there are multiples of the same integer, they are counted in duplicate.

**INPUT 2**

5 3 3

1 5 2 2 4

**OUTPUT 2**

4

The integers JOI writes down are:

* 5, for (l,r) = (1,3)
* 2, for (l,r) = (1,4)
* 2, for (l,r) = (1,5)
* 5, for (l,r) = (2,4)
* 4, for (l,r) = (2,5)
* 4, for (l,r) = (3,5)

Out of these, the L (= 3) th smallest integer is 4.

**INPUT 3**

6 2 9

1 5 3 4 2 4

**OUTPUT 3**

4

**INPUT 4**

6 2 8

1 5 3 4 2 4

**OUTPUT 4**

3