Special Numbers

Problem Statement

For each positive integer n, denote by d(n) the number of positive divisors of n. A positive integer n is said to be special if there is no k<n with d(k)=d(n). Compute the sum of all special numbers no greater than N.

Input: A single integer N.

Output: A single integer: the sum of all special numbers no greater than N.

Constraints: $1 \leq N \leq 100000$.

Sample Input:
4

Sample Output:
7

Explanation:
3 is not special since d(2)=d(3)=2. 1, 2, and 4 are special, so the answer is 1+2+4=7.

Message Relay

Farmer John's N cows (1 <= N <= 1000) are conveniently numbered from 1..N. Using an old-fashioned communicating mechanism based on tin cans and strings, the cows have figured out how to communicate between each-other without Farmer John noticing.

Each cow can forward messages to at most one other cow: for cow i, the value F(i) tells you the index of the cow to which cow i will forward any messages she receives (this number is always different from i). If F(i) is zero, then cow i does not forward messages.

Unfortunately, the cows have realized the possibility that messages originating at certain cows might ultimately get stuck in loops, forwarded around in a cycle forever. A cow is said to be "loopy" if a message sent from that cow will ultimately get stuck in a loop. The cows want to avoid sending messages from loopy cows. Please help them by counting the total number of FJ's cows that are not loopy.

INPUT FORMAT:
* Line 1: The number of cows, N.
* Lines 2..1+N: Line i+1 contains the value of F(i).

SAMPLE INPUT:
5
0
4
1
5
4

INPUT DETAILS:
There are 5 cows. Cow 1 does not forward messages. Cow 2 forwards messages to cow 4, and so on.

OUTPUT FORMAT:
* Line 1: The total number of non-loopy cows.

SAMPLE OUTPUT:
2

OUTPUT DETAILS:
Cow 1 is not loopy since she does not forward messages. Cow 3 is also not loopy since she forwards messages to cow 1, who then does not forward messages onward. All other cows are loopy.

Perimeter

Farmer John has arranged N hay bales (1 <= N <= 50,000) in the middle of one of his fields. If we think of the field as a 1,000,000 x 1,000,000 grid of 1 x 1 square cells, each hay bale occupies exactly one of these cells (no two hay bales occupy the same cell, of course).

FJ notices that his hay bales all form one large connected region, meaning that starting from any bale, one can reach any other bale by taking a series of steps either north, south, east, or west onto directly adjacent bales. The connected region of hay bales may however contain "holes" -- empty regions that are completely surrounded by hay bales.

Please help FJ determine the perimeter of the region formed by his hay bales. Note that holes do not contribute to the perimeter.

INPUT FORMAT:

* Line 1: The number of hay bales, N.
* Lines 2..1+N: Each line contains the (x,y) location of a single hay bale, where x and y are integers both in the range 1..1,000,000. Position (1,1) is the lower-left cell in FJ's field, and position (1000000,1000000) is the upper-right cell.

SAMPLE INPUT:

```
8
10005 200003
10005 200004
10008 200004
10005 200005
10006 200003
10007 200003
10007 200004
10006 200005
```

INPUT DETAILS:

The connected region consisting of hay bales looks like this:

```
XX
X XX
XXX
```

OUTPUT FORMAT:

* Line 1: The perimeter of the connected region of hay bales.

SAMPLE OUTPUT:
14

OUTPUT DETAILS:
The length of the perimeter of the connected region is 14 (for example, the left side of the region contributes a length of 3 to this total). Observe that the hole in the middle does not contribute to this number.

Tractor

One of Farmer John's fields is particularly hilly, and he wants to purchase a new tractor to drive around on it. The field is described by an N x N grid of non-negative integer elevations (1 <= N <= 500). A tractor capable of moving from one grid cell to an adjacent cell (one step north, east, south, or west) of height difference D costs exactly D units of money.

FJ would like to pay enough for his tractor so that, starting from some grid cell in his field, he can successfully drive the tractor around to visit at least half the grid cells in the field (if the number of total cells in the field is odd, he wants to visit at least half the cells rounded up). Please help him compute the minimum cost necessary for buying a tractor capable of this task.

INPUT FORMAT:
* Line 1: The value of N.
* Lines 2..1+N: Each line contains N space-separated non-negative integers (each at most 1 million) specifying a row of FJ's field.

SAMPLE INPUT:
5
0 0 0 3 3
0 0 0 0 3
0 9 9 3 3
9 9 9 3 3
9 9 9 9 3

INPUT DETAILS:
FJ's farm is a 5 x 5 grid. The elevations in the first row are 0, 0, 0, 3, and 3, and so on.

OUTPUT FORMAT:
* Line 1: The minimum cost of a tractor that is capable of driving around at least half of FJ's field.

SAMPLE OUTPUT:
3

OUTPUT DETAILS:
A tractor of cost 3 is capable of moving between elevation 0 and elevation 3, so it can visit the block of cells at zero elevation as well as the block of cells at elevation 3. Together, these represent at least half of FJ's farm.

Partitioning the Farm

Farmer John's farm is divided into an N x N square grid of pastures (2 <= N <= 15). Right now, there is a fence around the outside of the farm, but cows can move freely from pasture to pasture.

Farmer John has decided to build fences to separate the cows from each other. Because of zoning laws, each fence must be a horizontal or vertical line going across the entire farm and fences cannot go through pastures. Farmer John only has enough money to build at most K fences (1 <= K <= 2N - 2).

Farmer John wants to build the fences in order to minimize the size of the largest resulting group of cows (two cows are in the same group if they can reach each other without going through any fences). Given the current number of cows in each pasture, help Farmer John compute the size of the largest group of cows if he builds the fences optimally.

INPUT FORMAT:
* Line 1: Two integers, N and K
* Lines 2..1+N: There are N numbers per line, describing the cows in each
  pasture for one row of the farm (there are at least 0 and at most
  1000 cows in each pasture)

SAMPLE INPUT:
3 2
1 1 2
1 1 2
2 2 4

OUTPUT FORMAT:
* Line 1: The minimum possible size of the largest group of cows.

SAMPLE OUTPUT:
4

OUTPUT DETAILS:
Farmer John should build fences between columns 2 and 3 and between rows 2 and 3, which creates 4 groups each with 4 cows.