

D. Three-dimensional Turtle Super Computer

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A super computer has been built in the Turtle Academy of Sciences. The computer consists of $n \cdot m \cdot k$ CPUs. The architecture was the parallelepiped of size $n \times m \times k$, split into $1 \times 1 \times 1$ cells, each cell contains exactly one CPU. Thus, each CPU can be simultaneously identified as a group of three numbers from the layer number from 1 to n , the line number from 1 to m and the column number from 1 to k .

In the process of the Super Computer's work the CPUs can send each other messages by the famous turtle scheme: CPU (x, y, z) can send messages to CPUs $(x + 1, y, z)$, $(x, y + 1, z)$ and $(x, y, z + 1)$ (of course, if they exist), there is no feedback, that is, CPUs $(x + 1, y, z)$, $(x, y + 1, z)$ and $(x, y, z + 1)$ cannot send messages to CPU (x, y, z) .

Over time some CPUs broke down and stopped working. Such CPUs cannot send messages, receive messages or serve as intermediates in transmitting messages. We will say that CPU (a, b, c) *controls* CPU (d, e, f) , if there is a chain of CPUs (x_i, y_i, z_i) , such that $(x_1 = a, y_1 = b, z_1 = c)$, $(x_p = d, y_p = e, z_p = f)$ (here and below p is the length of the chain) and the CPU in the chain with number i ($i < p$) can send messages to CPU $i + 1$.

Turtles are quite concerned about the denial-proofness of the system of communication between the remaining CPUs. For that they want to know the number of critical CPUs. A CPU (x, y, z) is *critical*, if turning it off will disrupt some control, that is, if there are two distinctive from (x, y, z) CPUs: (a, b, c) and (d, e, f) , such that (a, b, c) controls (d, e, f) before (x, y, z) is turned off and stopped controlling it after the turning off.

Input

The first line contains three integers n, m and k ($1 \leq n, m, k \leq 100$) — the dimensions of the Super Computer.

Then n blocks follow, describing the current state of the processes. The blocks correspond to the layers of the Super Computer in the order from 1 to n . Each block consists of m lines, k characters in each — the description of a layer in the format of an $m \times k$ table. Thus, the state of the CPU (x, y, z) is corresponded to the z -th character of the y -th line of the block number x . Character "1" corresponds to a working CPU and character "0" corresponds to a malfunctioning one. The blocks are separated by exactly one empty line.

Output

Print a single integer — the number of critical CPUs, that is, such that turning only this CPU off will disrupt some control.

Examples

input
2 2 3 000 000 111 111
output
2

input
3 3 3 111 111 111 111

111 111
111 111 111
output
19

input
1 1 10 0101010101
output
0

Note

In the first sample the whole first layer of CPUs is malfunctional. In the second layer when CPU (2, 1, 2) turns off, it disrupts the control by CPU (2, 1, 3) over CPU (2, 1, 1), and when CPU (2, 2, 2) is turned off, it disrupts the control over CPU (2, 2, 3) by CPU (2, 2, 1).

In the second sample all processors except for the corner ones are critical.

In the third sample there is not a single processor controlling another processor, so the answer is 0.