## Problem 5 – Double Downs

|  |
| --- |
| 0 0 1 0 1 0 1 0  0 1 0 0 1 0 0 1 |

You are given a **number N and N integers**. Write a program to **count** all **couples** **of** **bits** between every two integers (**num[0]** and **num[1]**, **num[1]** and **num[2]**, …, **num[N-2]** and **num[N-1]**). You should count **3 kinds** of couples: **vertical,** **left-diagonal** and **right-diagonal** like at the example on the right. Every “**1”** **bit** can be part of **multiple** **couples.** Check the comments in the examples to understand your task better.

### Input

The input data should be read from the console.

* The number **n** stays at the first line.
* At each of the next **n** lines **n** integers are given, each at a separate line.

The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

The output should be printed on the console. It should consist of exactly **3** lines:

* At the **first** **line** print the count of the **right diagonal** couples.
* At the **second** **line** print the count of the **left diagonal** couples.
* At the **third** **line** print the count of the **vertical** couples.

### Constraints

* The number **n** will be an **integer** number in the range [2…100].
* The **n numbers** will be integers in the range [0…2 147 483 647].
* Allowed working time for your program: 0.25 seconds.
* Allowed memory: 16 MB.

### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Output** | **Bit Representation** | **Comments** |
| 3  19  18  49 | 1  2  3 | 0 0 0 1 0 0 1 1  0 0 0 1 0 0 1 0  0 0 1 1 0 0 0 1 | 1 right-diagonal couples  2 left-diagonal couples  3 vertical couples |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 5  167  153  14  23  18 | 8  5  7 |  | 4  1  2  3  4 | 1  2  1 |  | 2  0  0 | 0  0  0 |