## D:\Za uchene\Програмиране\Soft Uni\C# Basics\My exam problems\chess.PNGProblem 4 – Chess Queens

We are given a chess board of size **N \* N**. The only figures we have on the   
chess board are two queens. The queen in chess can move in **horizontal**, **vertical** and **diagonal** directions.We are also given a **number D** whichrepresents the distance between the two queens. The distance is measured by **D squares away**. All positions on the chessboard are represented with numbers and letters **(a1, a2… a8, b1-b8, c1-c8, …, h1-h8)**. Example: if **N=16**, the numbers on the board will be represented with integers **(1-16)** and letters **(a-o)**. Your task is to find **all** **couples** of queens where the queens stay either on the **same** **vertical**, **horizontal** or **diagonal**, at **distance** **D**. See the diagram aside to understand your task better. The green queens meet the condition of **2** blocks away but the red queens aren’t.

### Input

The input data should be read from the console. It consists of 2 lines:

* The first line holds an integer number **N** representing the **width** and **height** of the **chess** **board**.
* The second line holds an integer number **D** representing the **distance** that we should be looking for.

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 as a sequence of chess position in the **format [quеen1X, quеen1Y - quеen2X, quеen2Y]**. The order of the output is **not important.** Each string should stay on a separate line. In case they are no valid positions print “**No valid position**”.

### Constraints

* The **numbers N** and **D** will be **integers** in the range [0…20].
* Allowed memory: 16 MB.

### Examples

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 3  1 | a1 - a3  a1 - c1  a1 - c3  a2 - c2  a3 - a1  a3 - c1  a3 - c3  b1 - b3  b3 - b1  c1 - a1  c1 - a3  c1 - c3  c2 - a2  c3 - a1  c3 - a3  c3 - c1 | 4  2 | a1 - a4  a1 - d1  a1 - d4  a2 - d2  a3 - d3  a4 - a1  a4 - d1  a4 - d4  b1 - b4  b4 - b1  c1 - c4  c4 - c1  d1 - a1  d1 - a4  d1 - d4  d2 - a2  d3 - a3  d4 - a1  d4 - a4  d4 - d1 | 8  7 | No valid positions |