# Problem 4 – Bulls and Cows

All we love the “Bulls and Cows” game (<http://en.wikipedia.org/wiki/Bulls_and_cows>). Given a 4-digit **secret number** and a 4-digit **guess number** we say that we have **b** **bulls** and **c** **cows** when we have **b** matching digits on their right positions (bulls) and **c** matching digits on different positions (cows). Examples are given below:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Secret number | 1 | 4 | 8 | 1 | Bulls = 1  Cows = 2 |  | Secret number | 2 | 2 | 4 | 9 | Bulls = 0  Cows = 3 |
| Guess number | 8 | 8 | 1 | 1 | Guess number | 9 | 9 | 2 | 4 |

Given the secret number and the number of bulls and cows your task is to write a program to **find all matching guess numbers** in increasing order.

## Input

* The input data should be read from the console.
* It will consist of exactly 3 lines. At the first line there **the secret number** will be given. At the second line the number of bulls **b** will be given. At the third line the number of cows **c** will be given.
* The input data will always be valid and in the format described. There is no need to check it explicitly.

## Output

* The output data should be printed on the console
* It should consist of a single line holding all matched guess numbers, given in increasing order, separated by single space.

## Constraints

* The **secret number** will always consist of exactly 4 digits, each in the range [1…9].
* The numbers **b** and **c** will be in the range [0…9].
* Time limit: 0.15 seconds.
* Allowed memory: 4 MB.

## Examples

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| 2228  2  1 | 1222 2122 2212 2232 2242 2252 2262 2272 2281 2283 2284 2285 2286 2287 2289 2292 2322 2422 2522 2622 2722 2821 2823 2824 2825 2826 2827 2829 2922 3222 4222 5222 6222 7222 8221 8223 8224 8225 8226 8227 8229 9222 | 1234  3  0 | 1134 1214 1224 1231 1232 1233 1235 1236 1237 1238 1239 1244 1254 1264 1274 1284 1294 1334 1434 1534 1634 1734 1834 1934 2234 3234 4234 5234 6234 7234 8234 9234 | 1234  3  1 | No |

Attribution: this work may contain portions from the "[C# Part I](https://telerikacademy.com/Courses/Courses/Details/81)" course by Telerik Academy under [CC-BY-NC-SA](http://creativecommons.org/licenses/by-nc-sa/3.0/deed.en_US) license.