
 Home

 Scoreboard

 Solutions

 You advanced to **Round 2!**

**Score:** 28 / 100 points

**Rank:** 4 174th out of 12 692

[View Certificate](#)

PROBLEMS

Weak Typing

☒ A1: Chapter 1 10 pt

☐ A2: Chapter 2 14 pt

☐ A3: Chapter 3 21 pt

☒ **B: Traffic Control** 18 pt

☐ C: Blockchain 37 pt

 FAQ

 My Clarifications

Problem B: Traffic Control

Submit for Practice

18 points

 Accepted

Problem

My Submissions

Solution

In the rapidly growing towns of Silicon Valley, traffic congestion is becoming a major problem. The governor has contracted Travis, the rock star traffic engineer, to come up with some solutions.

A town consists of  $N$  horizontal streets and  $M$  vertical avenues, making up  $N * M$  intersections. Each intersection has some wait time, which is an integer number of seconds between 1 and 1,000, inclusive. The town's wait times can thus be represented as a grid with  $N$  rows and  $M$  columns of integers.

A *route* within the grid consists of a sequence of one or more intersections, such that each pair of consecutive intersections are either horizontally or vertically adjacent to one another. The *duration* of a route is the sum of all of its intersections' wait times.

Travis is tasked with assigning valid wait times for all  $N * M$  intersections. However, he must be careful in his choices. Routes too slow will lead to congestion, and routes too fast will lead to accidents. After careful calculations, Travis has determined that that an optimal grid must simultaneously satisfy the following two benchmark conditions based off the four corners of the town:

1. The duration of the fastest route starting in the top-left intersection and ending in the bottom-right one is exactly  $A$  seconds, and
2. the duration of the fastest route starting in the top-right intersection and ending in the bottom-left one is exactly  $B$  seconds.

Please help Travis generate any such valid grid, or determine that no such grid exists.

Constraints

$1 \leq T \leq 40$

$2 \leq N, M \leq 50$

$1 \leq A, B \leq 1000$

The sum of  $N * M$  across all towns is at most 40,000.

Input

Input begins with an integer  $T$ , the number of towns that Travis is contracted for. For each town, there is a single line containing the 4 space-separated integers  $N$ ,  $M$ ,  $A$ , and  $B$ .

Output

For the  $i$ th town, first print a line containing "*Case #i:* " followed by either the string "Possible" or "Impossible". If possible, this should be followed  $N$  rows of  $M$  space-separated integers each, comprising your chosen grid of intersection wait times in seconds.

Sample Explanation

In the first town, the following is one possible grid for which the minimum durations of routes from the top-left to the bottom-right intersection, as well as from the top-right to the bottom-left intersection, are each equal to  $333 + 333 + 333 = 999$  seconds (as required):

```
333 333
333 333
```

In the second town, the following is one possible grid satisfying the requirements:

```
5 3 1
3 4 3
```

For example, the fastest route from the top-left to the bottom-right intersection involves moving right, then right, then down, for a total duration of  $5 + 3 + 1 + 3 = 12$  seconds. Meanwhile, the fastest route from the top-right to the bottom-left intersection has a duration of  $1 + 3 + 4 + 3 = 11$  seconds.

For each of the first 3 towns, other outputs would also be accepted.

Sample Input



```
4
2 2 999 999
2 3 12 11
4 3 6 6
50 50 1 1
```

Sample Output



```
Case #1: Possible
333 333
333 333
Case #2: Possible
5 3 1
3 4 3
Case #3: Possible
1 1 1
1 2 1
1 2 1
1 1 1
Case #4: Impossible
```

Submit for Practice