Keyboards

In her rare free time, Lea sometimes builds mechanical keyboards for her computer. Today she wants to experiment with a new fancy type of mechanical switches. She has two layout ideas to try out and compare. To avoid losing any switches during reassembly it would be really nice if the key count could be the same for both! But is it feasible?

For each layout there are some keys in special positions (for example, the spacebar or shift or control), and then there is a repeating part with some number of keys per column that can be stretched as long as desired. To make sure she gets enough exercise even when she has to work on an article all day long, Lea embraces the idea of the keyboards being long enough to require a bit of jogging.

Or maybe Lea wants an ortholinear keyboard, with 7 special-case keys in the middle and repeatable columns of four keys each?

Given the numbers of the special keys and the sizes of repeatable columns, can you say whether the project is feasible?

Input

The first line of the input contains an integer t. t test cases follow.

Each test case is a line containing five integers s_1, c_1, s_2, c_2, n. Here s_1 and c_1 are the numbers of special keys and the number of keys per column in the first design, s_2 and c_2 are the corresponding numbers for the second design, and n is the minimal acceptable number of keys.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x being the minimum number of keys that is at least n and can be used to implement both designs. If there is no such x, output "impossible". Each line of the output should end with a line break.

Note that if there are enough special keys, it might be possible to implement a design using zero columns.

Constraints

- $1 \le t \le 20$
- $0 \le n \le 10^{15}$
- $0 \le s_1, s_2 \le 2 \cdot 10^{15}$
- $1 \le c_1, c_2 \le 2 \cdot 10^{15}$
- If x exists, it is no larger than $4 \cdot 10^{18}$

Sample Input 1

```
5
10 7 8 5 40
0 6 0 10 20
0 2 1 2 0
1 998 1 997 900
0 2 0 3 999
```

Sample Output 1

```
Case #1: 73
Case #2: 30
Case #3: impossible
Case #4: 995007
Case #5: 1002
```