Problem A - Stashing stuff

Department of Computer Science

University of Cape Town, South Africa

CSC1015F - problem set

So you've made it to Cape Town and are 'in res'—or if you were already in the area, perhaps in a student dig anyway. Trying to settle into your own room, you have to put some of your belongings into the cupboards, in the most space-efficient way. Now, you *could* do it in a trial-and-error way, stuffing various things into the cupboard and out to try to fit something else in there, which is way to tiresome to do. The budding computer scientist you are—or just lazy—you've found a smarter way: measure the dimension of the (rectangular) cupboard, and then lazily from the couch, measure up some of your belongings and let a computer program tell you whether it fits!

For now, we only consider the rectangular objects of all your stuff, with a length, width, and hight. For best stacking of objects, the sides of the object must be parallel to the sides of the cupboard. The cupboard's dimensions are 50cm by 50cm by 50cm.

Input Input starts with an integer T ($T \leq 100$), which indicates the number of test cases. Each of the next T line contains three integers L, W and H ($1 \leq L, W, H \leq 50$) denoting the length, width and height of a rectangular object that might fit in the cupboard.

Output For each test case, output a single line. If the object fits in the cupboard in any orientation having the sides of the object in parallel to the sides of the cupboard, this line will be '`Case #: good`', otherwise it will be '`Case #: bad`'. In your output, `#` will be replaced by the case number. Please see the sample input and sample output for exact format.

Sample input

```
2
20 20 20
1 2 51
```

Sample output

```
Case 1: good
Case 2: bad
```