700 Date Bugs

There are rumors that there are a lot of computers having a problem with the year 2000. As they use only two digits to represent the year, the date will suddenly turn from 1999 to 1900. In fact, there are also many other, similar problems. On some systems, a 32-bit integer is used to store the number of seconds that have elapsed since a certain fixed date. In this way, when 2^{32} seconds (about 136 Years) have elapsed, the date will jump back to whatever the fixed date is.

Now, what can you do about all that mess? Imagine you have two computers C_1 and C_2 with two different bugs: One with the ordinary Y2K-Bug (i.e. switching to $a_1 := 1900$ instead of $b_1 := 2000$) and one switching to $a_2 := 1904$ instead of $b_2 := 2040$. Imagine that the C_1 displays the year $y_1 := 1941$ and C_2 the year $y_2 := 2005$. Then you know the following (assuming that there are no other bugs): the real year can't be 1941, since, then, both computers would show the (same) right date. If the year would be 2005, y_1 would be 1905, so this is impossible, too. Looking only at C_1, we know that the real year is one of the following: 1941, 2041, 2141, etc. We now can calculate what C_2 would display in these years: 1941, 1905, 2005, etc. So in fact, it is possible that the actual year is 2141.

To calculate all this manually is a lot of work. (And you don't really want to do it each time you forgot the actual year.) So, your task is to write a program which does the calculation for you: find the first possible real year, knowing what some other computers say (y_i) and knowing their bugs (switching to a_i instead of b_i). Note that the year a_i is definitely not after the year the computer was built. Since the actual year can't be before the year the computers were built, the year your program is looking for can't be before any a_i.

Input

The input file contains several test cases, in which the actual year has to be calculated. The description of each case starts with a line containing an integer n ($1 \leq n \leq 20$), the number of computers. Then, there is one line containing three integers y_i, a_i, b_i for each computer ($0 \leq a_i \leq y_i < b_i < 10000$). y_i is the year the computer displays, b_i is the year in which the bug happens (i.e. the first year which can't be displayed by this computer) and a_i is the year that the computer displays instead of b_i.

The input is terminated by a test case with $n = 0$. It should not be processed.

Output

For each test case, output the line "`Case #`k`:`", where k is the number of the situation. Then, output the line "`The actual year is` z`.`", where z is the smallest possible year (satisfying all computers and being greater or equal to $u = \max_{i=1}^{n} a_i$). If there is no such year less than 10000, output "`Unkown bugs detected.`".

Output a blank line after each case.

Sample Input

```
2
1941 1900 2000
2005 1904 2040
2
1998 1900 2000
1999 1900 2000
0
```

Sample Output

```
Case #1:
The actual year is 2141.

Case #2:
Unknown bugs detected.
```