






























































































































































As everyone knows, immortal porpoises live on the other side of straight lines and eat live seagulls, although Federal Law prohibits transporting the gulls to them (but that’s another story).

The well-known scientist Leo O. Pisa studied immortal porpoises at great length, and made a number of interesting discoveries. He found, for example, that a single pair of porpoises brings forth a pod of 2 babies every year, and that it takes a full year for the babies to mature and start producing pods of their own. Therefore, if you start with one pair of newborn porpoises, at the end of the 1st year, you would still have one pair of porpoises (now mature and ready to procreate). At the end of the second year, you would have a nice new pod of newborns. But you would also have a pair of mature adults! Next year, they would have a second pod of newborns, but their 1st pod would be ready to start producing pods of their own.

Each year, all the porpoises that were around last year are still there. But in addition, all the porpoises who were around two years earlier have babies! However, to prevent the world from being overrun with immortal porpoises, whenever their numbers reach one billion (10^9) pairs or more, one billion of the pairs mysteriously disappear. Strangely enough, some years the population disappears completely. When that happens, in the next year a single pair miraculously appears. In fact, for any given year, Y , the number of living porpoise pairs is $fib(Y) \bmod 10^9$, where $fib(Y)$ is the Y -th value in the well-known *Fibonacci* sequence.

Suppose each picture  represents the number of pairs alive at the end of the year. Then our year-to-year Porpoise Population chart would look something like this:

Year	Pair Count	Porpoise Pairs Alive
1	1	
2	1	
3	2	 
4	3	  
5	5	    
6	8	       
7	13	               
8	21	                    
⋮	⋮	⋮
43	433494437	                     ...    
44	701408733	                     ...    
45	134903170	                     ...   
46	836311903	                     ...    

Your job is to write a program that determines the number of immortal porpoise pairs alive at the end of a given number of years, assuming that you start with a newborn pair at the beginning of the 1st year. Since immortal porpoises believe that the universe will exist for exactly $2^{48} = 281,474,976,710,656$ years, you need not deal with years later than that.

Input

The first line of input contains a single integer P , ($1 \leq P \leq 1000$), which is the number of data sets that follow. Each data set should be processed identically and independently.

Each data set consists of a single line of input consisting of two space-separated values. The first value is an integer, K , which is the data set number. The next value Y , ($1 \leq Y \leq 2^{48}$), is the number of years the porpoises have been alive.

Output

For each data set there is a single line of output. The line consists of the data set number, K , a single space, and the number of immortal porpoise pairs alive at the end of Y years.

Sample Input

11
1 1
2 2
3 8
4 20
5 46
6 60
7 3749999998
8 3749999999
9 3750000000
10 3750000001
11 281474976710656

Sample Output

1 1
2 1
3 21
4 6765
5 836311903
6 8755920
7 499999999
8 500000001
9 0
10 500000001
11 309764667