11421   Arranging Cards

Given n ($n \leq 50$) different cards, you're going to arrange them in a sequence so that adjacent cards always have different ranks.

Each card is represented by two characters: its rank (`2..9`, `T` for ten, `J` for Jack, `Q` for Queen, `K` for King, `A` for Ace) and suite (`S` for Spade, `C` for Club, `H` for Heart, `D` for Diamond). The whole sequence is represented by the concatenation of all the cards. For example, `6H7DJDKC` is a valid sequence, but `6H7D7SKC` is not, because there are two adjacent cards `7D` and `7S` with the same rank.

Given a positive integer k ($1 \leq k \leq 10^{18}$), find the lexicographically k-th smallest sequence.

Input

The input contains at most 1000 test cases. Each case begins with two integers n and k in the first line. The second line contains n cards separated by a single space. The case with $n = 0$ indicates the end of the input and should not be processed.

Output

For each case, print the case number and the answer. If there is no such sequence, print 'Not found' (without quotes). Insert a single space between each two adjacent cards to make it look better.

Sample Input

```
6 1
2S 3S 3C 4S 4C 4D
6 120
2S 3S 3C 4S 4C 4D
6 121
2S 3S 3C 4S 4C 4D
16 654321234567
2S 3S 4S 5S 2C 3C 4C 5C 2D 3D 4D 5D 2H 3H 4H 5H
0 0
```

Sample Output

```
Case 1: 2S 4C 3C 4D 3S 4S
Case 2: 4S 3S 4D 3C 4C 2S
Case 3: Not found
Case 4: 5D 4S 2D 5H 3S 4H 5S 2H 3D 2C 5C 4D 2S 3C 4C 3H
```