You're transmitting an n-bits unsigned integer k through a simulated network. The i-th bit counting from left is transmitted at time i (e.g. 4-bit unsigned integer 5 is transmitted in this order: 0-1-0-1). The network delay is modeled as follows: if a bit is transmitted at time i, it may arrive at as early as $i + 1$ and as late is $i + d + 1$, where d represents the maximal network delay. If more than one bit arrived at the same time, they could be received in any order.

For example, if you're transmitting a 3-bit unsigned integer 2 (010) for $d = 1$, you may receive 010, 100 (first bit is delayed) or 001 (second bit is delayed).

Write a program to find the number of different integers that could be received, and the smallest/largest ones among them.

Input

The input contains several test cases. Each case consists of three integers n, d, k ($1 \leq n \leq 64$, $0 \leq d \leq n, 0 \leq k < 2^n$), the number of bits transmitted, the maximal network delay, and the integer transmitted. The last test case is followed by a single zero, which should not be processed.

Output

For each test case, print the case number and the number of different integers that could be received, followed by the minimal and maximal one among them.

Sample Input

```
3 0 2
3 1 2
10 2 888
7 3 107
0
```

Sample Output

```
Case 1: 1 2 2
Case 2: 3 1 4
Case 3: 25 490 984
Case 4: 19 47 122
```