Hmm! Here you are asked to do a simple sorting. You will be given N numbers and a positive integer M. You will have to sort the N numbers in ascending order of their modulo M value. If there is a tie between an odd number and an even number (that is their modulo M value is the same) then the odd number will precede the even number. If there is a tie between two odd numbers (that is their modulo M value is the same) then the larger odd number will precede the smaller odd number and if there is a tie between two even numbers (that is their modulo M value is the same) then the smaller even number will precede the larger even number.

For remainder value of negative numbers follow the rule of C programming language: A negative number can never have modulus greater than zero. E.g. `-100 MOD 3 = -1, -100 MOD 4 = 0`, etc.

Input

The input file contains 20 sets of inputs. Each set starts with two integers N ($0 < N \leq 10000$) and M ($0 < M \leq 10000$) which denotes how many numbers are within this set. Each of the next N lines contains one number each. These numbers should all fit in 32-bit signed integer. Input is terminated by a line containing two zeroes.

Output

For each set of input produce $N + 1$ lines of outputs. The first line of each set contains the value of N and M. The next N lines contain N numbers, sorted according to the rules mentioned above. Print the last two zeroes of the input file in the output file also.

Sample Input

```
15 3
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
0 0
```

Sample Output

```
15 3
15
9
3
6
12
13
7
1
4
10
11
5
2
8
14
0 0
```