Czech ACM Student Chapter
CTU in Prague, Dept. of Computer Science and Engineering
VŠB TU Ostrava, FEI, Dept. of Computer Science
SUT Bratislava, FEI, Dept. of Computer Science and Engineering

CTU Open Contest 2002

Defragmentation

`defrag.p`, `defrag.c`, `defrag.C`

For the maximal security, the special secure operation system is used in the Summit Headquarters. This OS uses also the special file system. In this file system all disk space is divided into N clusters of equal sizes, numbered by integers from 1 to N. Each file occupies one or more clusters in arbitrary areas of the disk. All clusters that are not occupied by files are considered to be free. A file can be read from the disk in the fastest way, if all its clusters are situated in the successive disk clusters in the natural order.

Rotation of the disk with constant speed implies that various amounts of time are needed for accessing its clusters. Therefore, reading of clusters located near the beginning of the disk performs faster than reading of the ones located near its end. Thus, all files are numbered beforehand by integers from 1 to K in the order of descending frequency of access. Under the optimal placing of the files on the disk the file number 1 will occupy clusters $1, 2, ..., S_1$, the file number 2 will occupy clusters $S_1 + 1, S_1 + 2, ..., S_1 + S_2$ and so on (S_i is the number of clusters occupied by the i-th file).

In order to place the files on the disk in the optimal way, cluster-moving operations are executed. One cluster-moving operation includes reading of one occupied cluster from the disk to the memory and writing its contents to some free cluster. After that, the first cluster is declared free, and the second one is declared occupied.

Your goal is to place the files on the disk in the optimal way by executing the minimal possible number of cluster-moving operations.

Input Specification

The input consists of several disk descriptions. The first line of the description contains two integers N and K separated by a space, $1 \le K < N \le 100000$. Then K lines follow, each of them describes one file. The description of the i-th file starts with the integer S_i that represents the number of clusters in the i-th file, $1 \le S_i \le N - K$. Then S_i integers follow separated by spaces, which indicate the cluster numbers occupied by this file on the disk in the natural order. Each of these numbers is between 1 and N, inclusive.

All cluster numbers in one disk description are different and there is always at least one free cluster on the disk. The input is terminated with the line containing two zeros in place of numbers N and K.

Output Specification

For each disk description, output exactly one line containing the sentence "`We need` M `move operations.`" where M is the minimal number of cluster-moving operations needed to place the files on the disk into the optimal clusters.

If the files on the disk are already placed in the optimal way, the output should only contain the sentence "`No optimization needed.`".

Sample Input

```
20 3
4 2 3 11 12
1 7
3 18 5 10
30 4
2 1 2
3 3 4 5
2 6 7
8 8 9 10 11 12 13 14 15
0 0
```

Output for Sample Input

```
We need 9 move operations.
No optimization needed.
```