# 2582 Galactic Breakup

After ruling a large chunk of the Milky Way for millennia, the Cosmic OBsolescent OLigarchy is finally breaking up into a collection of independent monarchies. COBOL is a very organized empire and takes the shape of a gigantic cube with dimensions n by m by k parsecs. (COBOL is also very secretive, so only a few know the exact values of n, m and k.) To facilitate the control of the empire it is partitioned into nmk smaller dominions, each 1 cubic parsec in size. These dominions are numbered as in the figure on the right.
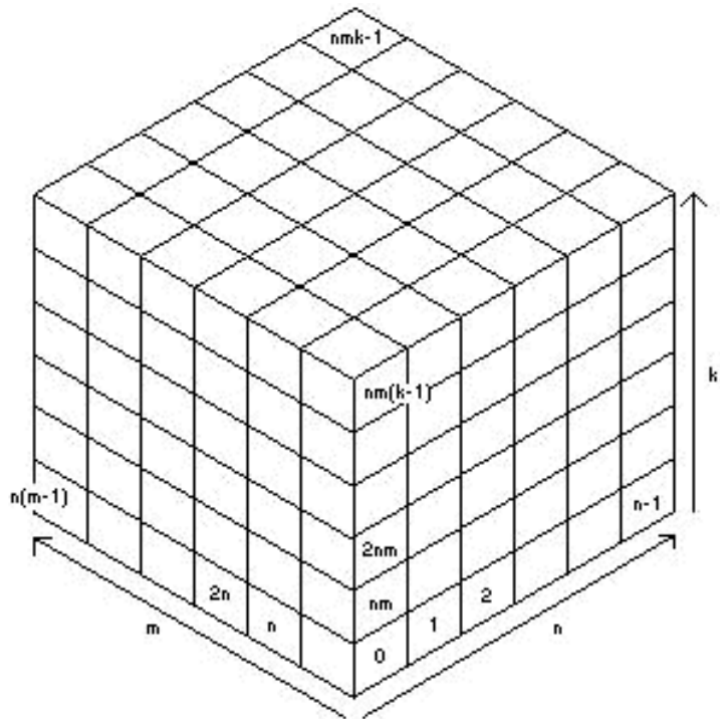
Each independent monarchy is a connected collection of one or more dominions (a dominion is connected to another if they share a face) and over a period of several imperial months, one monarchy per month will secede from the empire. Each secession begins at the first day of the month. One concern of COBOL is that during the breakup, various parts of the remaining empire may become disconnected from one another, which could hamper the administration of what's left of the empire. Your job is to determine the number of months of the breakup during which the empire is disconnected.

Input

Input will consist of multiple problem instances. The first line will contain a positive integer indicating the number of problem instances to follow. The first line of each problem instance will contain four integers: n m k l, where n, m and k are as described above, with $1 \leq n, m, k \leq 30$, and l is the number of independent monarchies which the empire is being divided into. Following this will be l lines defining the monarchies. Each will have the form p d_1 d_2 $d_3 \ldots d_p$, where p is the number of dominions making up the monarchy ($1 \leq p \leq 20$), and $d_1, \ldots, d_p$ are the dominions.

The monarchies are listed in the order in which they will secede from the empire.

Output

Output for each problem instance should consist of a single integer on a line, indicating the number of months which the empire was disconnected.

Sample Input

```
2
2 2 3 9
2 4 5
```

```
3 6 8 10
1 7
1 2
1 11
1 9
1 1
1 0
1 3
2 2 3 3
4 0 1 2 3
4 4 5 6 7
4 8 9 10 11
```

Sample Output

```
4
0
```