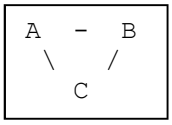
10. Stelios

Program Name: Stelios.java **Input File: stelios.dat**

Stelios is researching efficient ways to make connections within graphs and needs to determine the fewest number of steps it takes to get from one node to any other node. He knows about the direct connections within a network system, but needs to know about the indirect connections as well, ones that take two or more steps to reach.

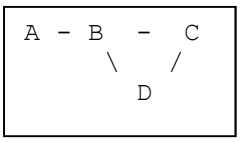
For example, in a simple graph as shown below, with nodes A, B and C, and direct connections between A and B, B and C, and A and C, it only takes one step to get from one node to any other node. The direct connections would be designated using alphabet pairs, like AB, BC and CA. The order of the letters in each pair does not matter. The link from A to B is the same as the link from B to A. The data could also be written BA, CB and AC and mean the same.



	A	B	C
A	0	1	1
B	1	0	1
C	1	1	0

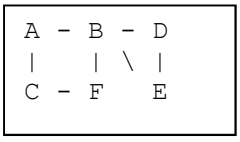
This graph can be represented by a connection matrix such as the one shown above. The value 1 means there is a direct connection from one node to the next. The first 1 in the top row means that you can get from node A to node B in one step. The next 1 indicates the direct connection from node A to C. The first 1 in the second row indicates the connection from B to A, and so on.

In a four-way graph like the one shown below, indicated by these data pairs - **AB BC CD BD** - it may take more than one step to get from node to another, as you can see in the connection matrix. To get from A to either C or D takes two steps, which is indicated by all the 2 values in the grid: A to C, A to D, C to A, and D to A.



	A	B	C	D
A	0	1	2	2
B	1	0	1	1
C	2	1	0	1
D	2	1	1	0

In this 6-node graph, the one-step connections are designated as AB, AC, CF, BF, BD, BE and ED, resulting in a graph and connection matrix as shown below.



	A	B	C	D	E	F
A	0	1	1	2	2	2
B	1	0	2	1	1	1
C	1	2	0	3	3	1
D	2	1	3	0	1	2
E	2	1	3	1	0	2
F	2	1	1	2	2	0

The longest connections in this example are between nodes C and D, or C and E, going either direction, each taking three steps.

Stelios needs your help to write a program to calculate the connection matrix given the number of nodes in the graph, and a number of alphabet pairs representing the one-step connections within the graph.

Input: Several sets of data, each set consisting of an integer N, followed by several uppercase alphabet pairs on the same line, as described and shown above, with single space separation.

(Stelios, cont)

Output: The N x N connection matrix as described above and shown in the sample outputs below, each value of the matrix representing the fewest steps it takes to get between nodes. Single space separation is required between all values on each line. Each grid is followed by a dashed line exactly the length taken by the grid, as shown in the sample output below.

Assumptions:

- 3 <= N <= 26
- The alphabet pairs will be derived from a set of sequential uppercase letters labeling the nodes of the graph, always starting with A and ending with the corresponding letter for the value of N, i.e. N = 5, last node letter is E, N = 6, last letter is F, etc. For example, a three-node graph will always use the letters A, B, and C, a four-node graph the letters A through D, a ten node graph the letters A through J, and so on.
- The most number of possible steps between any two nodes will be 9.

Sample Input:

```
3 AB BC CA
4 AB BC CD BD
6 AB AC CF BF BD BE ED
```

Sample Output:

```
0 1 1
1 0 1
1 1 0
-----
0 1 2 2
1 0 1 1
2 1 0 1
2 1 1 0
-------
0 1 1 2 2 2
1 0 2 1 1 1
1 2 0 3 3 1
2 1 3 0 1 2
2 1 3 1 0 2
2 1 1 2 2 0
-----------
```