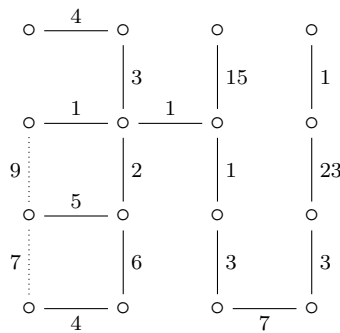


You are encouraged to work in groups of two or three. If you cannot find partners, you can work alone. Please submit **one** copy of the assignment; if you are working with partners, all names should appear on the assignment. Submit using **subversion**. Note that you will get a grade of zero if your program does not run.

Write a Python program that takes as input the description of a grid, and outputs its minimum cost spanning tree. An n -grid is a graph consisting of n^2 nodes, organized as a square array of $n \times n$ points. Every node may be connected to at most the nodes directly above and below (if they exist), and to the two nodes immediately to the left and right (if they exist). An example of a 4-grid is:



Note that it has $4^2 = 16$ nodes, and 17 edges. What is the largest number of edges that an n -grid may have? We have the following node-naming convention: we name the nodes from left-to-right, row-by-row, starting with the top row. Thus, our 4-grid is described by the following adjacency list:

$$4 : (1, 2; 4), (2, 6; 3), (3, 7; 15), (4, 8; 1), (5, 6; 1), (6, 7; 1), \dots \tag{1}$$

where the first integer is the grid size parameter, and the first two integers in each triple denote the two (distinct) nodes that describe (uniquely) an edge, and the third integer, following the semicolon, gives the cost of that edge.

Write a program `st.py` which, when given as input a list of triples—such as (1)—it checks whether the list describes a grid, and computes the minimum cost spanning tree of the grid. In our 4-grid example, the solid edges describe a minimum cost spanning tree. Also note that the edges in (1) are not required to be given in any particular order.

Your program `st.py` should take as input a file, say `graph.txt`, containing a list such as (1). For example, `2 : (1, 2; 9), (3, 4; 5), (2, 4; 6), (1, 3; 2)` and it should output, directly to the screen, a graph indicating the edges of a minimum cost spanning tree. The graph should be “text-based” with “*” describing nodes and “-” and “|” describing edges. In this example:

```
* *
| |
*-*
```