1. **PREREQUISITES TO RUNNING THE CODE:**
2. have a working python 3.10 installed.
3. have libraries networkx and matplotlib installed.
   * pip install networkx
   * pip install matplotlib
4. have "input.txt" empty file in same directory as the python executable.
5. go in terminal (bash on linux and cmd/powershell on windows) and change directory to where you store the executable.
6. run `py name\_of\_code.py` or `python name\_of\_code.py` or `python3 name\_of\_code.py`
7. input in input.txt is of the following format.
   * m,n m- number of lines of input following 2nd line.
8. n- number of rows in each line.
   * fragmentation array - contains values we use fragment/resolve the array.
   * next m lines are the input simplicial complex.
   * squares with top left vertice joined to bottom right
9. all in a grid pattern.

1. **MY CODE:**

NOTE IN MY CODE: I have given vertices negative numbers as labels and the positive number in graph represents its value (that is taken as input).

**STEP 1:**

insert the resolution values (in my code these cannot be the same as the vertices of the grid)

in between the grid vertices wherever their values lie between the values of the vertices forming the edge.

**STEP 2:**

Join all points in each simplice that have same values.

**STEP 3:**

**FRAGMENTATION**: We get all the fragments of each individual simplice.

**STEP 4:**

* labelling fragments based on minimum vertice on their boundary.
* vertically joining fragments of each simplice.
* then checking for all pairs of simplices if they share common edges.
* If they do then horizontally join the formed vertical graphs to get the dual graphs.

**Step 5:**

* for all edges in dual graph check if they have vertices with same value if so fuse the two vertices.
* construct the contour tree in this way.