**IDS-PROJECT**

**(considering max edges that a particular vertex can be a part of)**

**Explaining the graph-**

I have considered the graph to be an adjacency list using structure list to construct all possible vertices. Each vertex will have a list of graph nodes (structure graph node) connected by linked list.

In order to make sure that each vertex has random number of edges, I have added a structure variable for number of edges remaining. Only if number of edges remaining is greater than or equal to 1 for 2 particular nodes, will an edge be formed between those 2 nodes. Therefore, in order to reduce the time complexity, I made it such that if an edge is not formed (due to one of the nodes not having number of edges remaining greater than or equal to 1), then that edge will be deleted and we will progress to the next possible edge for the particular vertex.

**Explaining the 10000 and 3000 case** -

I have attached a few screenshots of the debugging process.

As you can see, the time complexity is very high due to every time a neighbour node has to be checked and made sure that it is possible to form an edge.

If not, then we must delete that edge and move on to the next.

This will take a lot of time to do considering we have to do this for many vertices each having a large number of edges possible.



