Graph Theory

-A set of vertexes and edges

-Graphs can be directed or undirected

-A path is a sequence of vertexes and edges

-Simple path and cycle paths

-A connected graph has all its vertexes connected

-Connected components are the connected sub-graphs

-A tree is a graph without cycles

-If a tree contains ending vertexes, then the tree has N – 1 edges, where N is the total number of vertex

-If we add a path from a vertex u to a vertex v, then the tree will have a cycle and it will become a graph

-A forest is a set of trees

-In a forest we have N – C edges, where N is the total number of vertex in the forest, and C is the number of connected components

-We can check if a graph is a tree, by checking whether it has a cycle and checking whether all its components are connected (we can use DFS or BFS)

-If a problem statement says that there are no parallel edges, or there are at most one road, then we have a graph without parallel edges

-Ways to represent a graph:

-Adjacency matrix

-Adjacency list

-DSU (Disjoint Set Union):

-Allows two operations:

-same\_set(v, u)

-unite(v, u)

-Each set can be represented in memory as a tree

-same\_set(v, u) can be implemented as:

-return get(v) == get(u), where get is equal to:

-get(v):

while(parent(v) ¡= v):

v = p(v)

return v

…….

-The best way to check i fan algorithm is efficient, is by using the Big Oh Notation, based on a function created from the constraints of the problema

-The algorithm to know the shortest path of a graph with edges of length “1” is Breadth First Search (BFS)