Project 3 :

Forest of tree:

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2012/lecture-notes/MIT6_046JS12_lec16.pdf>

stl list

<https://en.cppreference.com/w/cpp/container/list>

I need to do like I did for the engine project with the hash table and the trees – make an index which inherent from the father index. Each child will contain the disjoint set of trees or lists.

for the list implementation:

I will have a vector (can be also list of lists) of list. Each time I will add a new list if there is no connection.

Make set will add a list of two element to the vector.

The find function will do a for loop thought the vector and each time will search in the list. If find, it will return the location of the list in the vector and add it to the related list, **then** delete it form the vector.

forest of trees:

I will have an array of pointers node(T). each will be assign at the beginning to null. The location of each element is according to the value of the string. For 2 letter it will be 26\*[0] + [1]. I need to look what is the ascii value for each letter first

I will have a node which contain value, rank (if parent it is 0) , pointer to parent.

Each time when inserting I will assign the parent, if no parent, it itself.

Maintaining vector of pairs (root,num of elements) or I can keep an counter in each node and just have a vector of roots

For loop to find parent until parent->data == data. Once it find the top parent assign the parent to it.