INFO6205 –Assignment3

Manasa Bhat

Section 5

NU ID: 001032278

Main Task Solution

1a). Implement height-weighted Quick Union with Path Compression.

I have implemented methods as below:

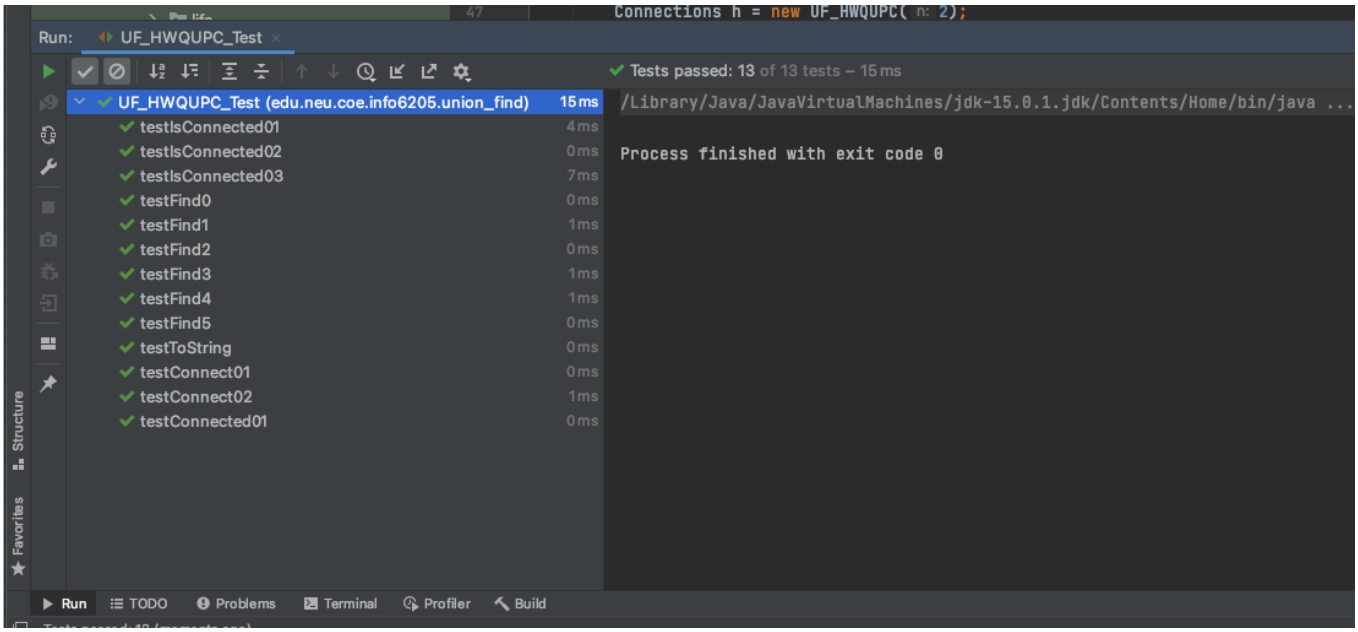
i) doPathCompression(int i): updated parent to value of grandparent.

ii) find(int p): return the root of p.

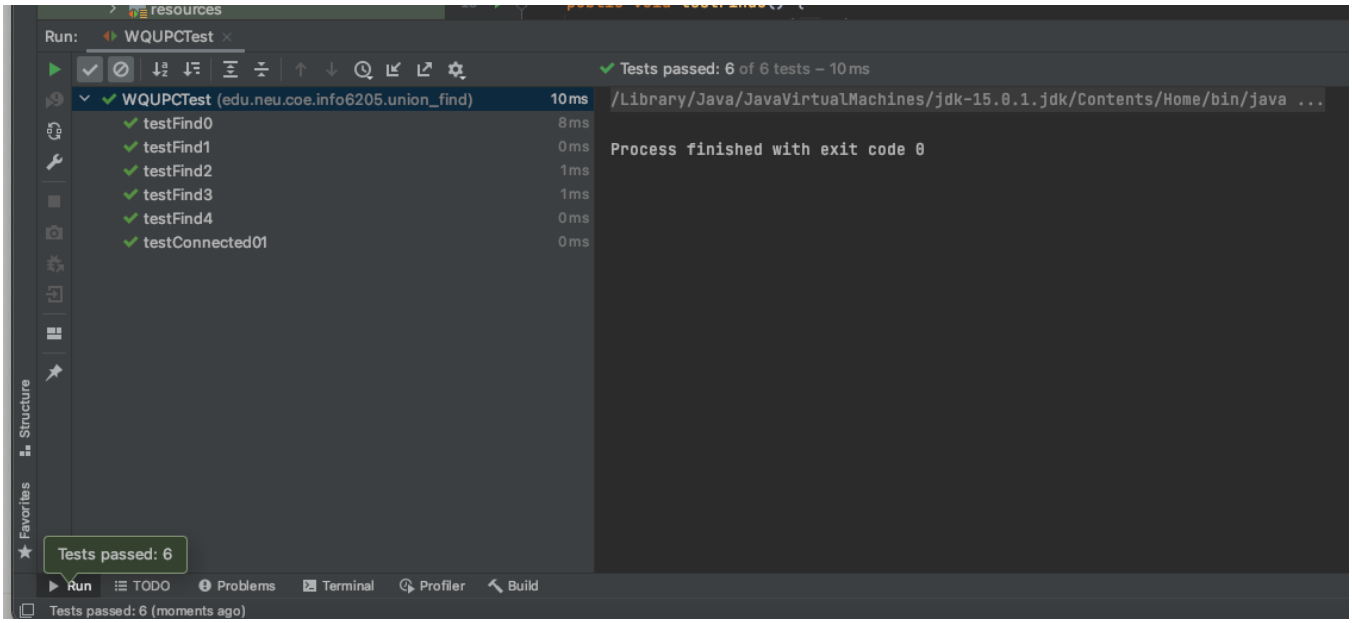
iii) mergeComponents(int i, int j): added logic to merge shorter root to taller one.

1b) All Unit Test cases passed. PFB the screenshots:

UF_HWQUPC_Test Output



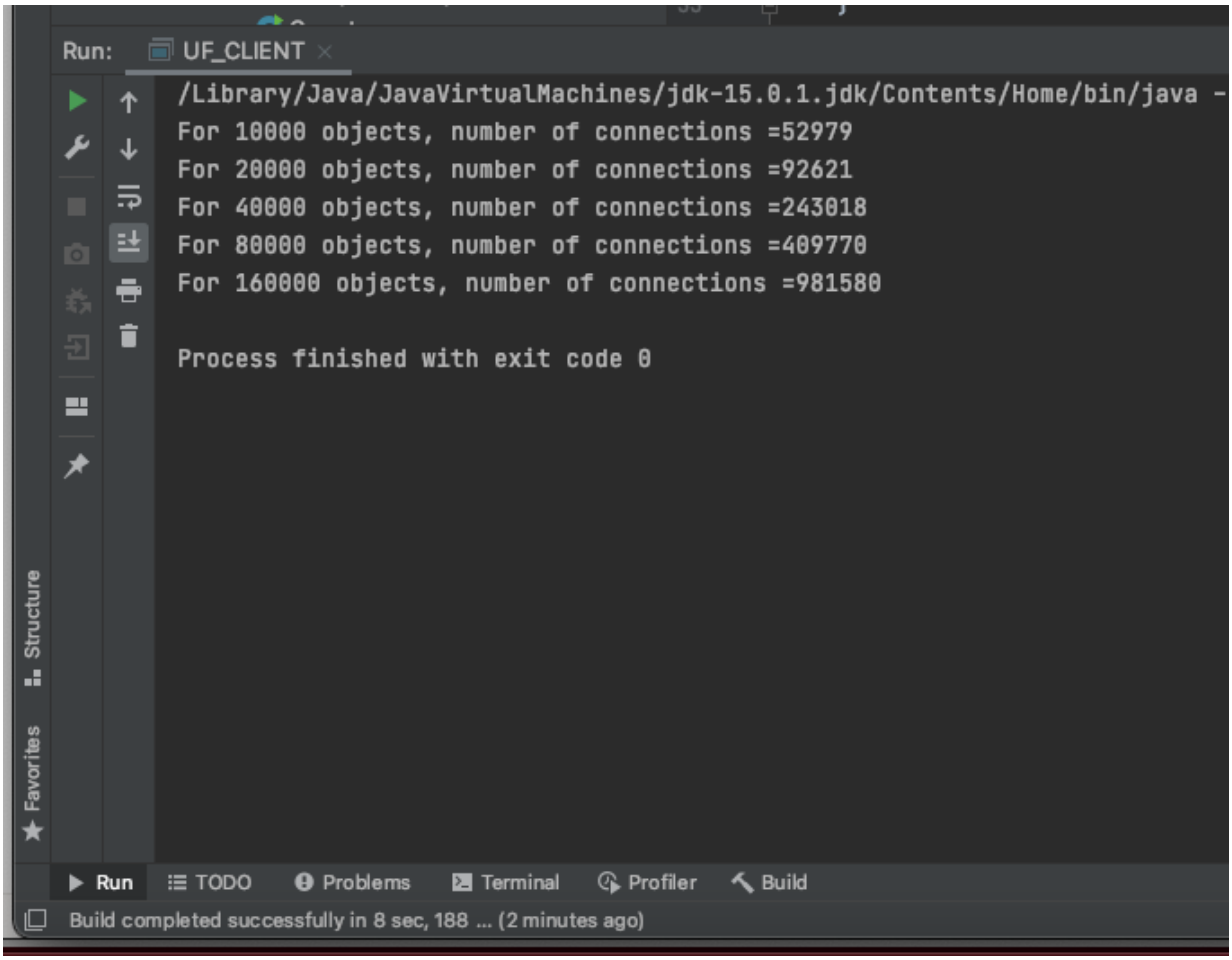
WQUPCTest Output:



2. Develop UF_CLIENT.

Implemented UF_HWQUPC in a new program called UF_CLIENT.java (**INFO6205-master 2/src/main/java/edu/neu/coe/info6205/union_find/UF_CLIENT.java)** that takes input as number of objects and returns the number of connections. As per instructions, I have created a main method that takes pre defined n values (10000, 20000, 40000, 80000, 160000) , calls count() method and prints the number of connections.

Evidence of runs:



3. Deducing relationship between number of objects(n) the number of pairs (m) generated to accomplish this (i.e. to reduce the number of components from n to 1).

Experiment Observations:

I conducted several runs for different n values to check the relation. In all the runs I could see there was an increase in the number of pairs required as the n value increases. Also, the number of pairs needed to reduce components to 1 was 4 to 5 times more than the number of objects taken.

Below are the observations:

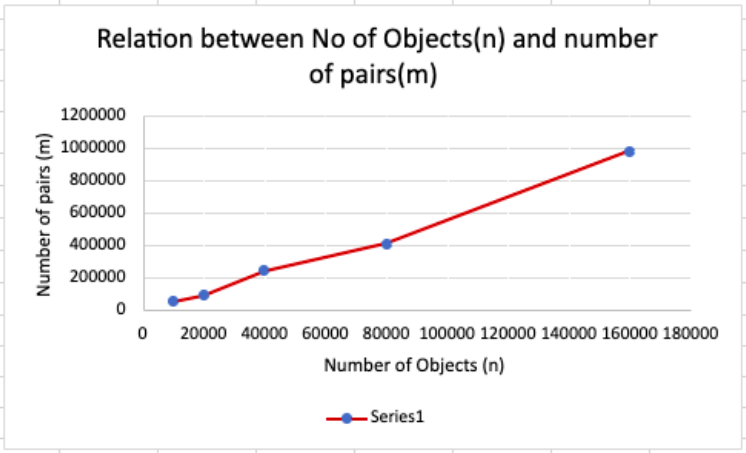
- Number of pairs formed(m) increases vastly as the number of objects (n) increases.
- Time taken to connect larger number of objects (Example 160000 objects) is far more than lower number of objects(Eg: 10000).
- On plotting the graph I could see there is almost a linear relationship between number of objects and connecting pairs needed.

- We could have a linear relation by taking nlog to base 2 of n value.
- The value of number of pairs needed(m) is greater than 1/3rd n logn but less than 1/4th of n logn.

Supporting Evidence

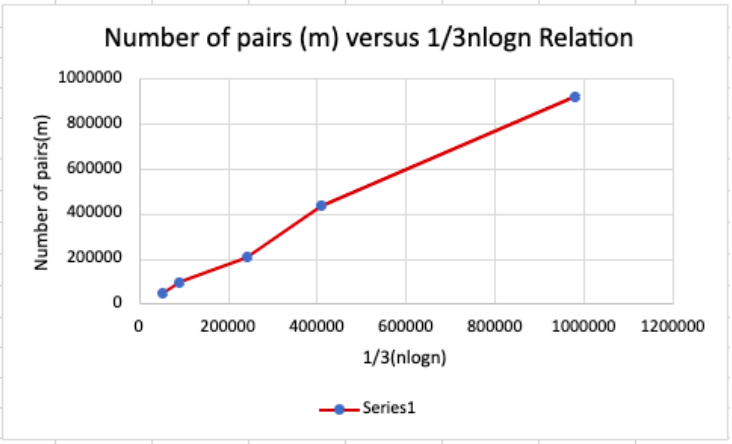
Below is the Number of Objects versus the connecting pairs relation plotted on a graph:

Objects N	Connections M
10000	52979
20000	92621
40000	243018
80000	409770
160000	981500



I got a more linear plot on taking 1/3rd nlogn with m number of pairs. Below is the plot depicting relation between number of pairs needed(m) and 1/3rd nlogn where n is the number of objects.

Objects N	1/3NlogN
52979	44292
92621	95251
243018	203836
409770	434338
981500	922011



As seen in the plots, I could deduce the relation of number of pairs needed (m) for reducing n components to 1 will be **O(1/3nlogn)**.

Conclusion

From the test results and observations, it can be concluded that number of pairs increases drastically with increase in number of objects. Time taken for components to reduce to 1 will depend on the number of objects taken.

Relationship of number of pairs needed to reduce components from n objects to 1 would be O(1/3nlogn).

Where n is number of objects and log is taken to base of 2.