NAME:SHREYA BALIGA
NU ID:002795178

ASSIGNMENT-4

TASK:

Step 1: Implement height-weighted Quick Union with Path Compression. Check for all the unit test cases

Step 2: Using the implementation of UF_HWQUPC, develop a UF ("union-find") client that takes an integer value n from the command line to determine the number of "sites." Generates random pairs of integers between 0 and n-1, calling connected() to determine if they are connected and union() if not. Loop until all sites are connected then print the number of connections generated. Package the program as a static method count() that takes n as the argument and returns the number of connections; and a main() that takes n from the command line, calls count() and prints the returned value or create a main program that doesn't require any input and runs the experiment for a fixed set of n values. Show evidence of the run(s).
Step 3: Determine the relationship between the number of objects (*n*) and the number of pairs (*m*)
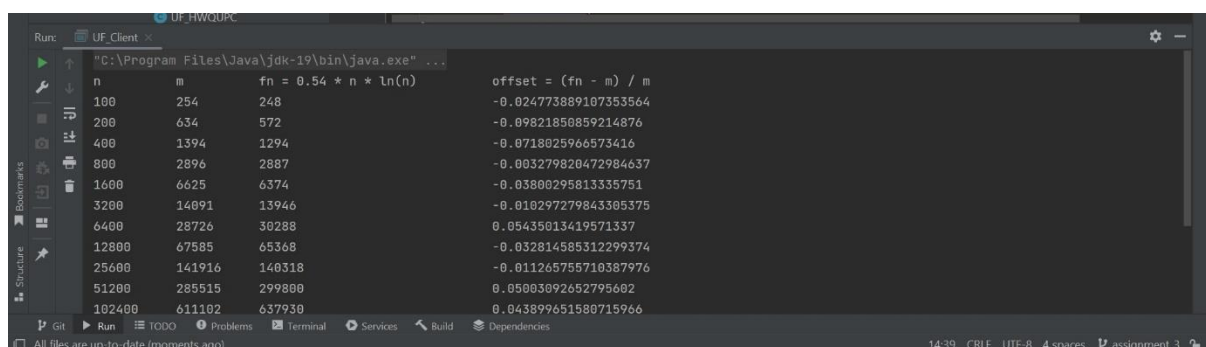
Relationship Conclusion:

Based on the data produced, it is evident that the number of pairs(m) created is proportional to the number of objects(n) supplied as input:

$$M=c*n*\log n$$
$$M \sim n\log n$$

OUTPUT:

The result below depicts the number of pairs(m), number of objects(n), function and offsetValue.

Evidence:

The below table and the graph depicts the relationship between the number of objects(n) and number of pairs(m) for different sets of values for both n and m.

number of objects (n)	Number of pairs generated(m)
100	277
200	534
400	1353
800	3188
1600	6958
3200	13713
6400	30301
12800	65037
25600	132349
51200	296898
102400	601016
204800	1270618
409600	2773753
819200	5940808
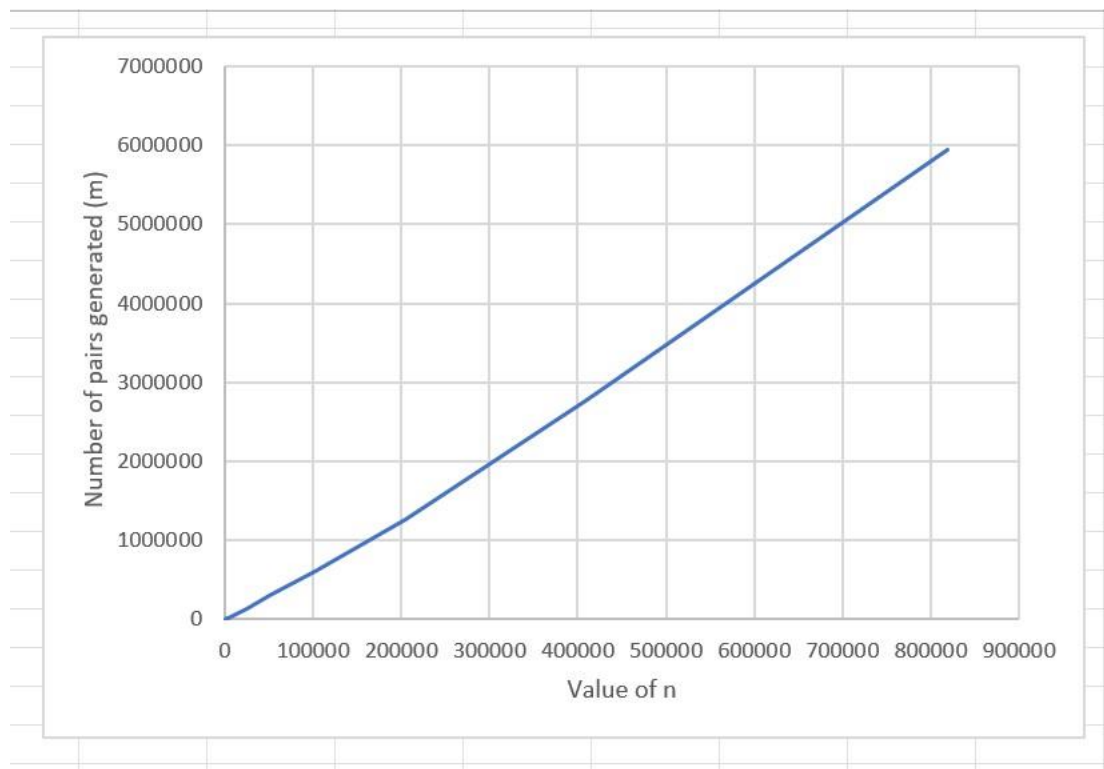
Passed Unit Test Cases:
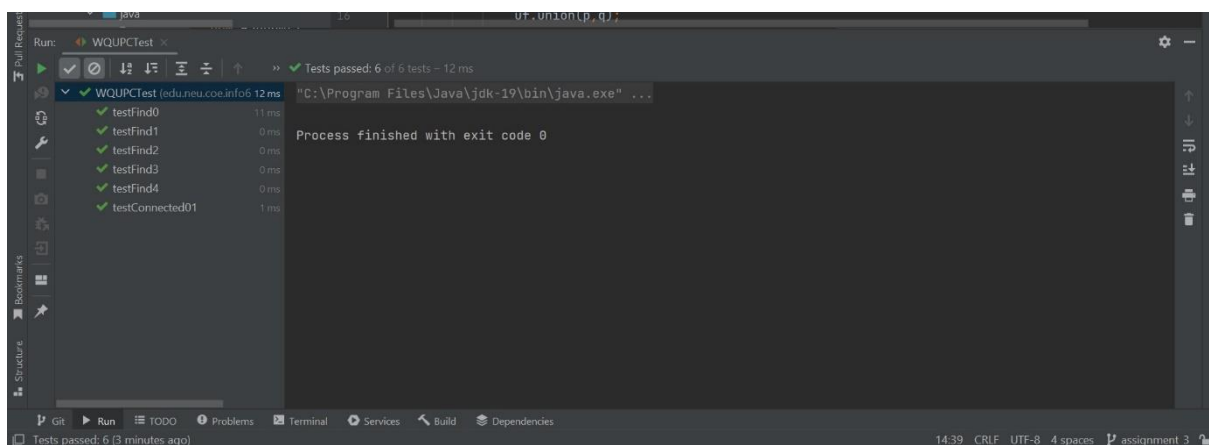
Source code:

The source code for this assignment is available on my repository:
https://github.com/ShreyaBaliga2408/Union_Find/tree/uf