Program Structures and Algorithms
Spring 2023(SEC –1)
Assignment 3 : Height-weighted Quick Union with Path Compression

NAME: Shubham Sharad Bagal
NUID: 002708621

Task:
The task entails writing a Java class that implements a height-weighted Quick Union with Path compression (UF HWQUPC). This class should include methods for performing union and connected operations. The second step is to create a union-find client that generates random pairs of integers, then calls the connected() and union() methods until all sites are connected. The client should return the number of generated connections. To reduce the number of components from n to 1, the final step is to determine the relationship between the number of objects and the number of pairs generated. Based on the observations, the conclusion should be justified.

Relationship Conclusion:

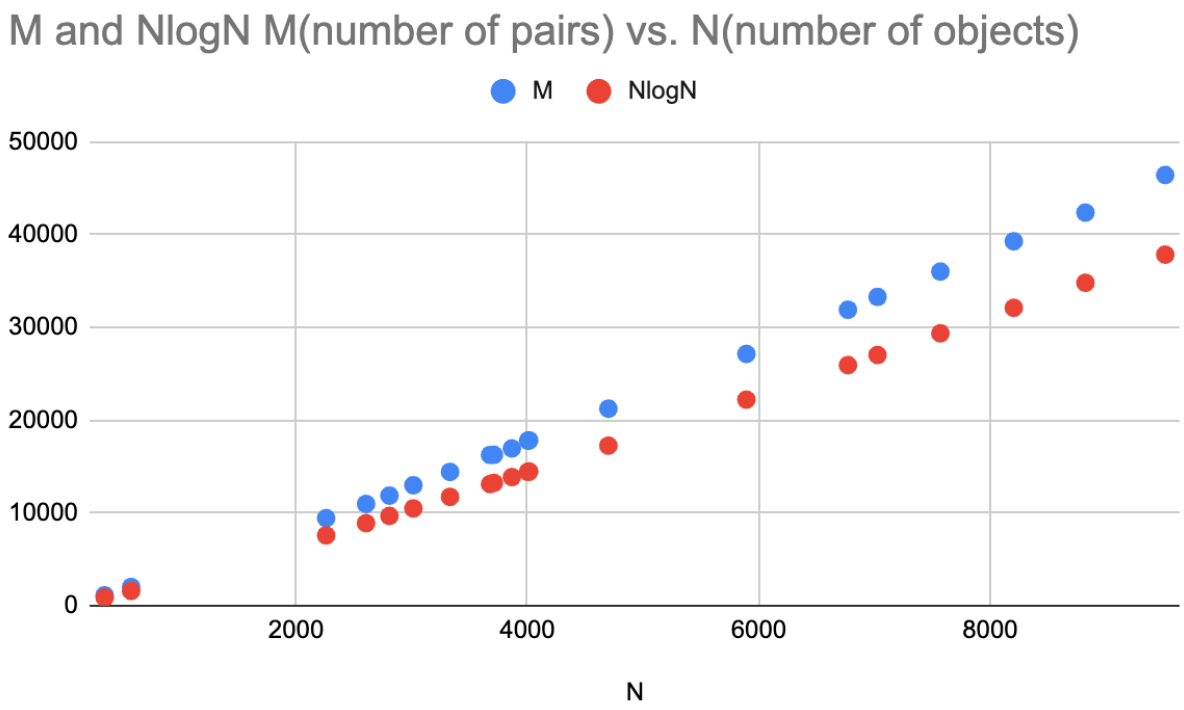
In general, the UF HWQUPC algorithm with path compression and height-weighted union has a near linear time complexity with a number of pairs of proportional linearithmic.

It is important to note that this is an average-case scenario that may vary depending on the specific implementation, data structure used, and input. It does, however, provide a broad overview of the relationship between the number of nodes and the number of pairs.

Evidence to support that conclusion:

N	M	NlogN
3866	16953	13868.35431
574	2021	1583.615426
6771	31910	25937.3502
3710	16268	13242.3772
4700	21240	17258.85993
3677	16254	13110.32008
4015	17830	14468.79748
2605	10971	8898.17913
345	1121	875.5475878
9514	46434	37850.14698
4006	17787	14432.46005
3331	14425	11733.6961
2260	9442	7580.285072
8205	39287	32115.01479
2807	11870	9679.216452
7570	36021	29364.75581
8824	42383	34816.55237
7027	33292	27031.25246
5893	27147	22218.59265
3014	12987	10486.13775

Graphical Representation:



Unit Test Screenshots: