Name: Rushikesh Deore
NUID: 002766913

Task: (a) Implement height-weighted Quick Union with Path Compression. For this, you will flesh out the class UF_HWQUPC. All you have to do is to fill in the sections marked with // TO BE IMPLEMENTED … // …END IMPLEMENTATION.

Program output:

Below is the screenshot of the test cases getting passed, ran through Junit in Eclipse.
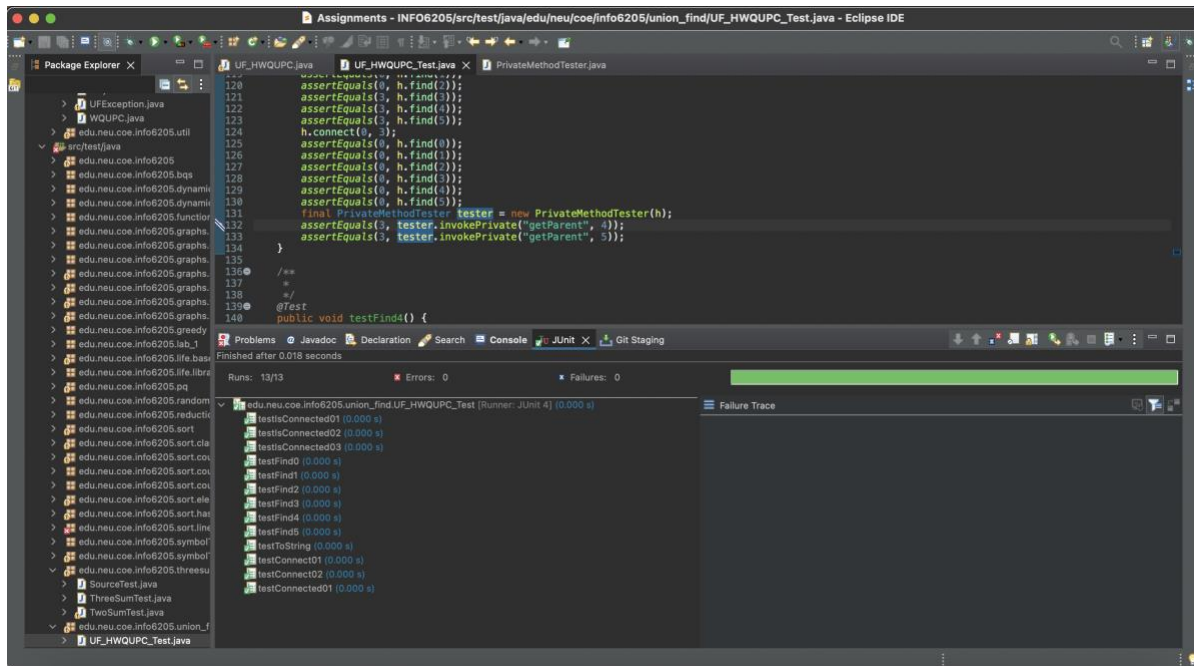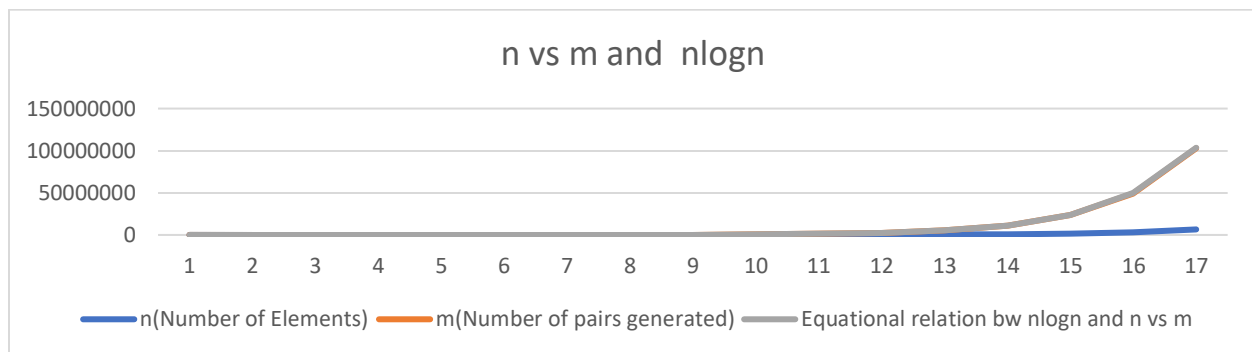
Graph of n (number of elements) vs m (number of connections to make it from n to 1):
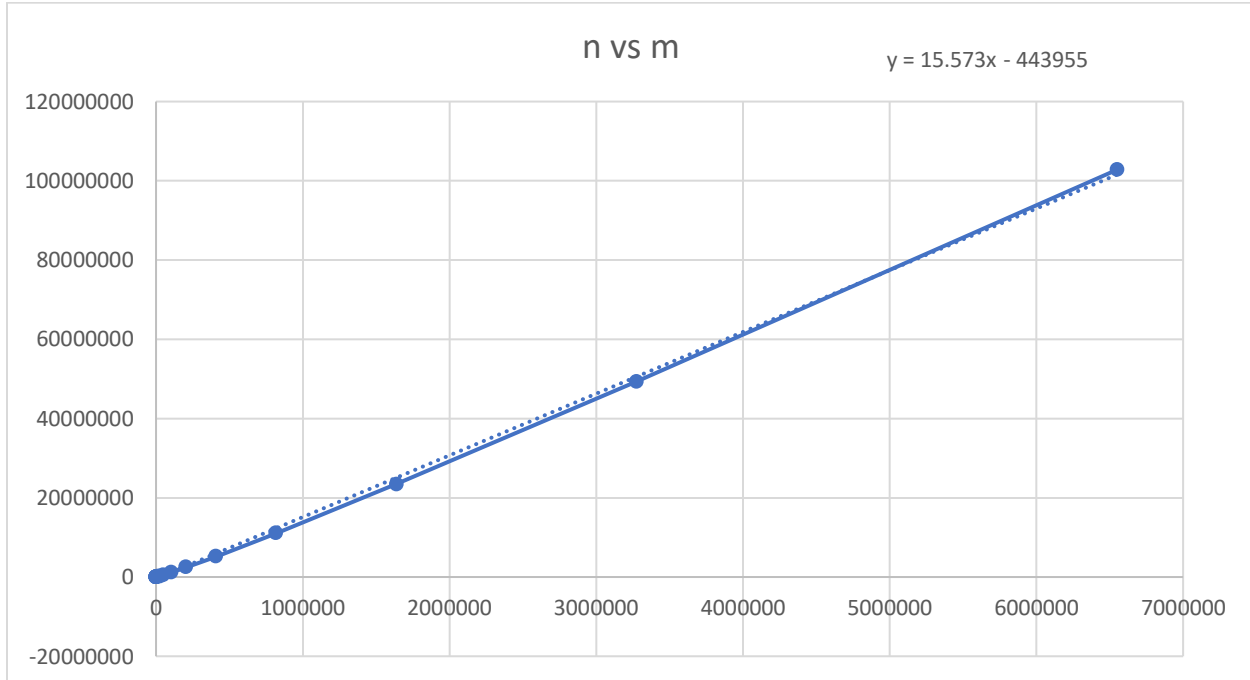
Graph of n*log(n) vs m (number of pairs obtained from code)

nlogn vs m

$y = 1.0006x - 11080$

Graph of n*log(n) vs m and values of n:

n vs m and nlogn

Legend: n(Number of Elements) — m(Number of pairs generated) — Equational relation bw nlogn and n vs m

Github Repo URL: https://github.com/rishdeore44/INFO6205.git
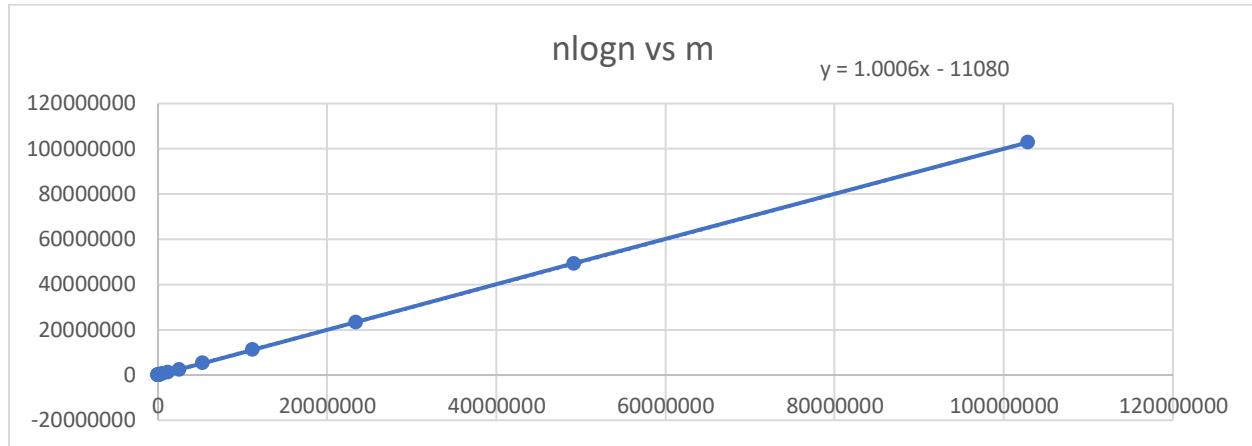
Conclusion:

The number of connections required to connect all the sites in a union-find implementation varies depending on the order of union operations. By running the program 50 times for different values of n (starting from 100 and doubling it), the relationship between the number of connections (m) and the number of elements in the union find (n) was found.

The relationship is estimated to be m = 1.0006 * n * ln(n), which means that on average, m is proportional to n * log(n). This estimate is based on the observation that a logarithmic number of connections are needed to connect all the sites in a balanced tree structure. As the number of elements in the union find increases, the actual number of connections becomes closer to the estimated value.

Also, if below table is to be looked at the equational relation, I am generating through n log n vs m and the actual n log n values are generating close values and I could say it specifies n*log n equation.

n(Number of Elements)	nlogn	m(Number of pairs generated)	Equational relation bw nlogn and n vs m
100	460.517019	300	463.2801207
200	1059.66347	850	1066.021454
400	2396.58582	2150	2410.965334
800	5347.68938	5044	5379.775518
1600	11804.4143	11083	11875.24074
3200	25826.8995	25155	25981.86088
6400	56089.9409	53951	56426.48057
12800	121052.166	119051	121778.4788
25600	259848.899	256125	261407.9927
51200	555186.934	542983	558518.0559
102400	1181352.14	1180527	1188440.253
204800	2504660.82	2501916	2519688.787
409600	5293234.73	5181618	5324994.138
819200	11154295.6	11076741	11221221.4
1638400	23444243.6	23420098	23584909.06
3276800	49159791.9	49336095	49454750.64
6553600	102862193	102860189	103479366.3