Program Structures and Algorithms
Spring 2023 (SEC –3)

NAME: Kaushik Gnanasekar
NUID: 002766012

Task: Assignment 4 (WQUPC)

Relationship Conclusion:

The implementation of height-weighted Quick Union with Path Compression takes *log(n)* time to merge components. For *n* sites to be merged, it'll take *nlog(n)* time. The path compression, flattens the height from nodes to roots reducing the overall time by a factor *k*, hence we get *k * nlog(n)*.

From the tests done for the number of objects (n) and the number of pairs (m) we find the relationship is:

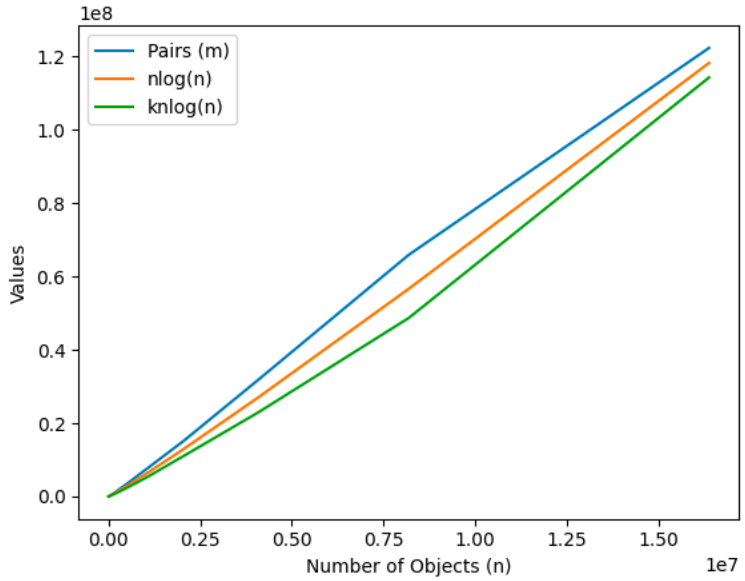
$m = nlog(n) * k$

Where k is some constant, which according to the tests done is 0.835458619.

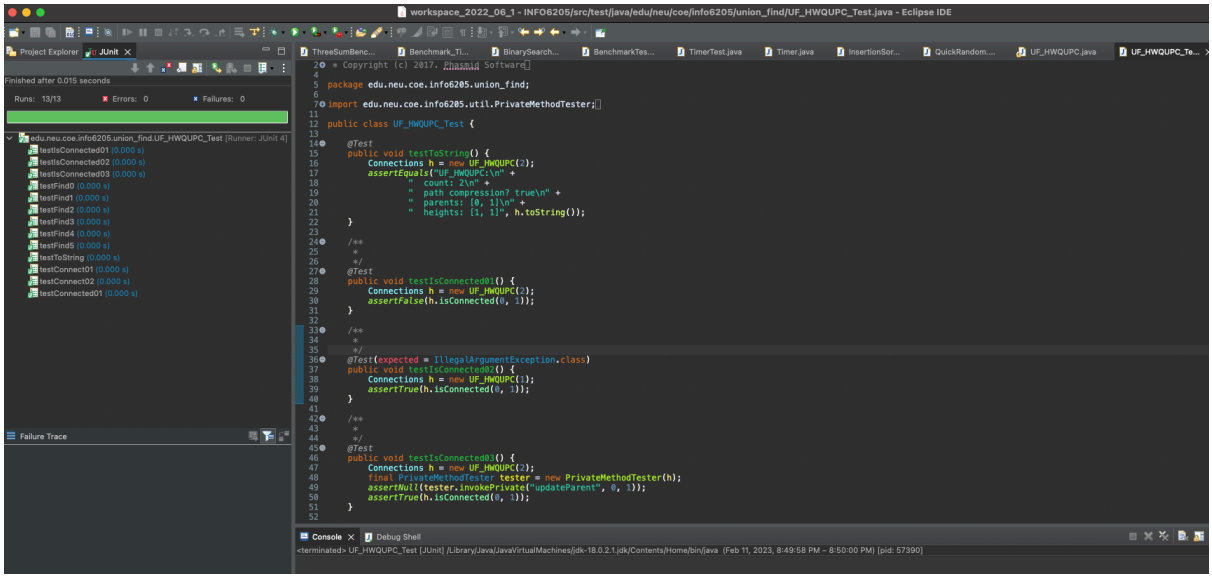
Evidence to support that conclusion:

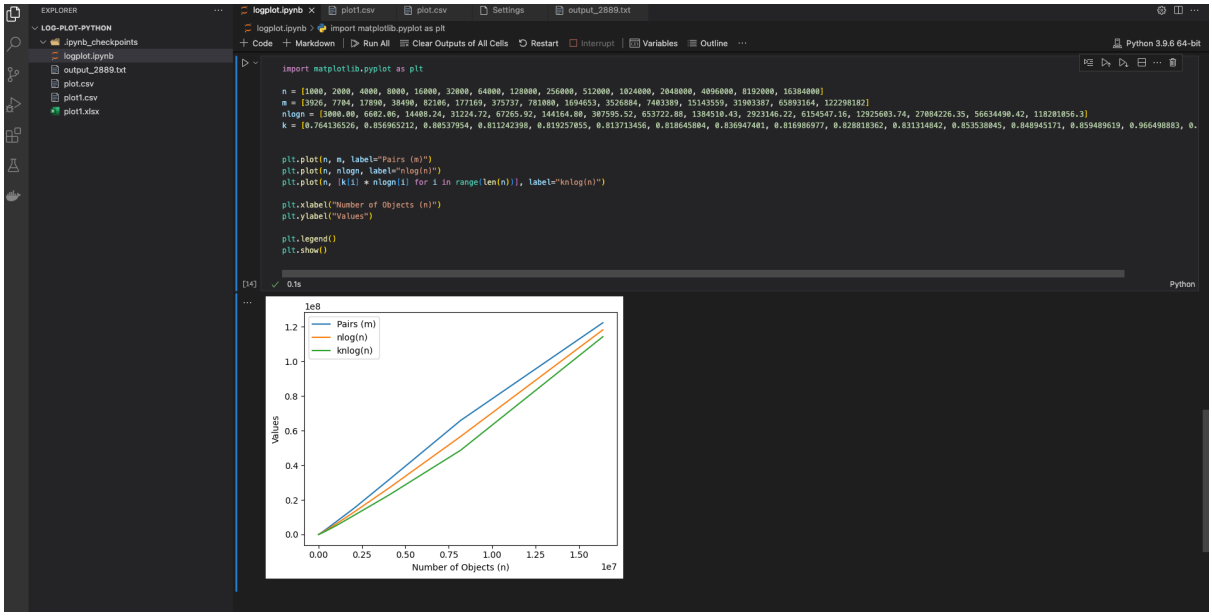
Number of Objects (n)	Number of Pairs (m)	nlog(n)	Constant (k)
1000	3926	3,000.00	0.764136526
2000	7704	6,602.06	0.856965212
4000	17890	14,408.24	0.80537954
8000	38490	31,224.72	0.811242398
16000	82106	67,265.92	0.819257055
32000	177169	144,164.80	0.813713456
64000	375737	307,595.52	0.818645804
128000	781080	653,722.88	0.836947401
256000	1694653	1,384,509.43	0.816986977
512000	3526884	2,923,146.22	0.828818362
1024000	7403389	6,154,547.16	0.831314842
2048000	15143559	12,925,603.74	0.853538045
4096000	31903387	27,084,226.35	0.848945171
8192000	65893164	56634490.42	0.859489619
16384000	122298182	118201056.3	0.966498883
			0.835458619

Graphical Representation:



Unit Test Screenshots:




EXPLORER
LOG-PLOT-PYTHON
.ipynb_checkpoints
logplot.ipynb
output_2889.txt
plot.csv
plot1.csv
plot1.xlsx
logplot.ipynb
plot1.csv
plot.csv
Settings
output_2889.txt
logplot.ipynb > import matplotlib.pyplot as plt
Code
Markdown
Run All
Clear Outputs of All Cells
Restart
Interrupt
Variables
Outline
Python 3.9.6 64-bit
import matplotlib.pyplot as plt
n = [1000, 2000, 4000, 8000, 16000, 32000, 64000, 128000, 256000, 512000, 1024000, 2048000, 4096000, 8192000, 16384000]
m = [3926, 7704, 17890, 38490, 82106, 177169, 375737, 781080, 1694653, 3526884, 7403389, 15143559, 31903387, 65893164, 122298182]
nlogn = [3000.00, 6602.06, 14408.24, 31224.72, 67265.92, 144164.88, 307595.52, 653722.88, 1384510.43, 2923146.22, 6154547.16, 12925603.74, 27084226.35, 56634490.42, 118201056.3]
k = [0.764136526, 0.856965212, 0.80537954, 0.811242398, 0.819257055, 0.813713456, 0.818645804, 0.836947401, 0.816986977, 0.828818362, 0.831314842, 0.853538045, 0.848945171, 0.859489619, 0.966498883, 0.
plt.plot(n, m, label="Pairs (m)")
plt.plot(n, nlogn, label="nlog(n)")
plt.plot(n, [k[i] * nlogn[i] for i in range(len(n))], label="knlog(n)")
plt.xlabel("Number of Objects (n)")
plt.ylabel("Values")
plt.legend()
plt.show()
[14]
0.1s
Python
1e8
Pairs (m)
nlog(n)
knlog(n)
Values
Number of Objects (n)
1e7