Deep Shah (002766755)

Program Structure & Algorithms

Spring 2023(Sec 03)

Assignment-4

Task:

There are three tasks to be performed.

Task list:

- (Task 1)
  (a) Implement height-weighted Quick Union with Path Compression. For this, you will flesh out the class UF_HWQUPC.
  (b) Check that the unit tests for this class all work. You must show "green" test results in your submission (screenshot is OK).
- (Task 2) Using your 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." Then 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 your 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. If you prefer, you can create a main program that doesn't require any input and runs the experiment for a fixed set of n values. Show evidence of your run(s).
- (Task 3) Determine the relationship between the number of objects (*n*) and the number of pairs (*m*) generated to accomplish this (i.e., to reduce the number of components from *n* to 1). Justify your conclusion in terms of your observations and what you think might be going on.

Relationship Conclusion:

Using the doubling method and considering different values ranging from 500 to 64000, plotting the No. of pairs generated for each value. Also taking the average of the No. of Paris generated after running it for 100 runs, we come with an equation that looks like.

No. of Pairs generated (m) = K * No. of Sites * $\log_2$(No. of Sites)

We get the equation $m = K * n\log_2(n)$
Here we can say that $K = m / n\log_2(n)$
Where K is a constant which is approx. equal to 0.53

Hence, we can prove that -

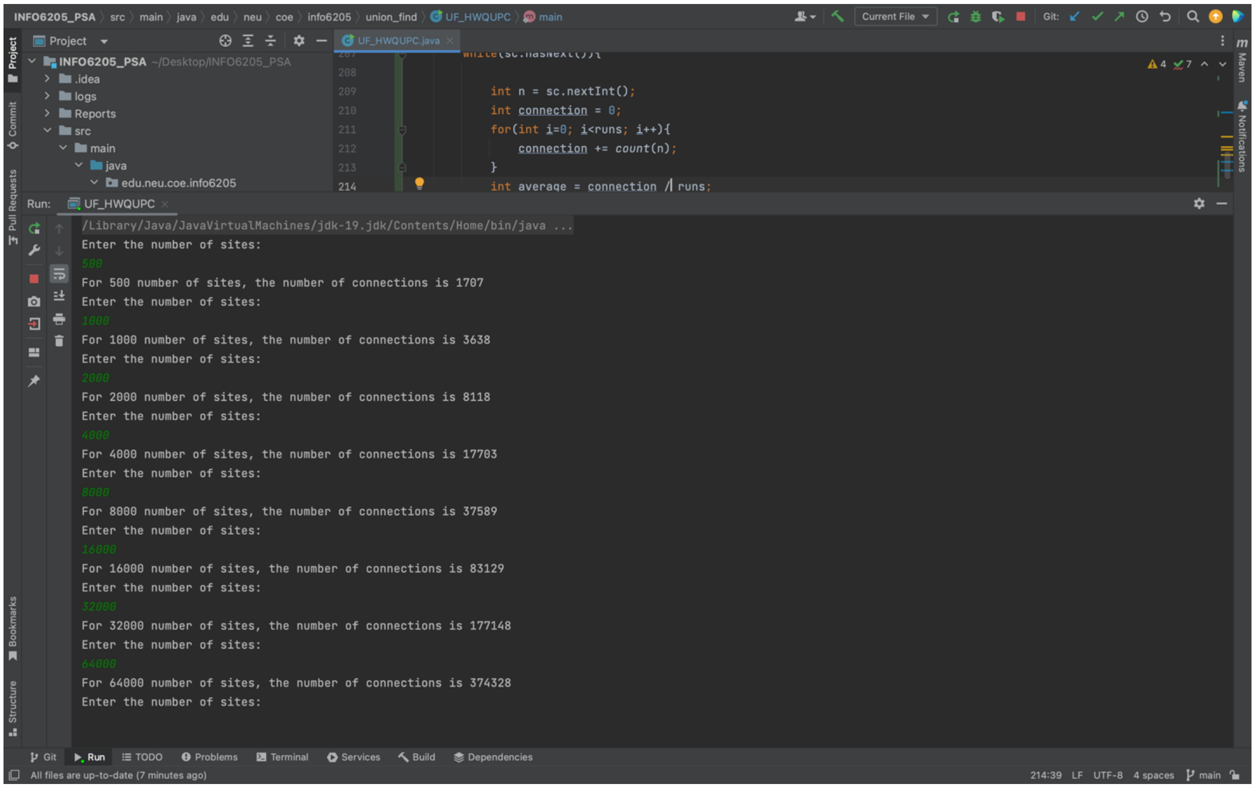
No. of Pairs generated (m) $\propto$ No. of Sites * $\log_2$(No. of Sites)
$m \propto n\log_2(n)$

Evidence to Support Conclusion:

To obtain the above relationship we created a main function in UF_HWQUPC. In this main function we created a count method that returns the count of the pairs that are connected. Hence obtaining the No. of Pairs generated for each value of n.

No. of Sites (n)	No. of Connections (m)	n log n	m / n log n
500	1707	3107.30405	0.54935081
1000	3638	6907.75528	0.52665444
2000	8118	15201.8049	0.53401554
4000	17703	33176.1986	0.53360544
8000	37589	71897.5746	0.52281319
16000	83129	154885.504	0.53671259
32000	177148	331951.718	0.53365592
64000	374328	708264.855	0.52851415

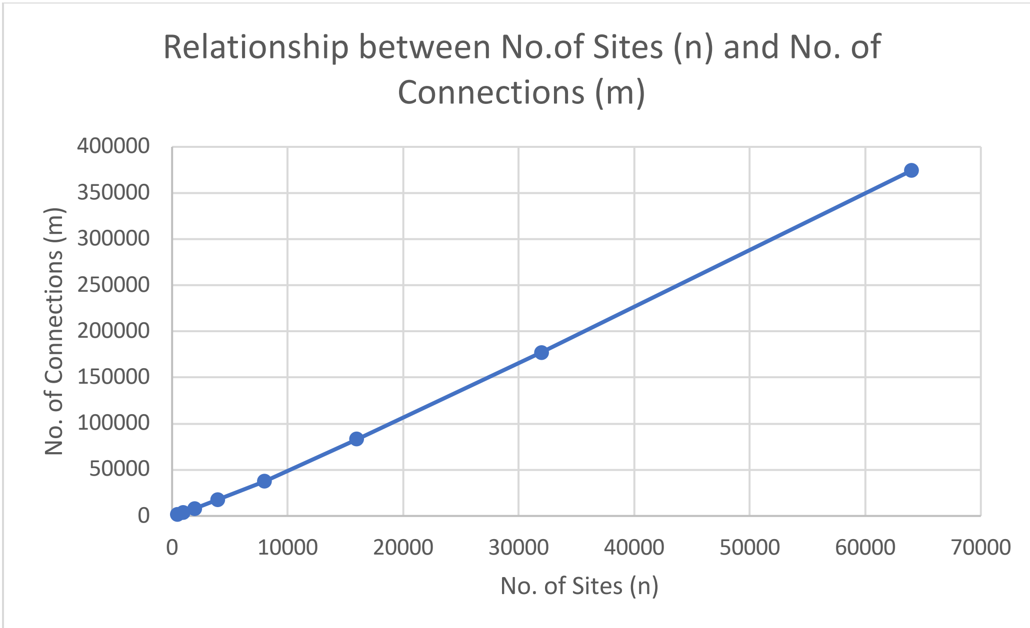
Output for No. of Pairs generated for different values of sites using the doubling method.



Graphical Representation:

As per the data plotted in the excel sheet for different No. of Sites, we get the following No. of Pairs generated.

No. of Sites (n)	No. of Connections (m)
500	1707
1000	3638
2000	8118
4000	17703
8000	37589
16000	83129
32000	177148
64000	374328



Unit Tests Result: