Venkata Raghu Teja Kumar Gollapudi (001529656)

Program Structures & Algorithms

Fall 2021

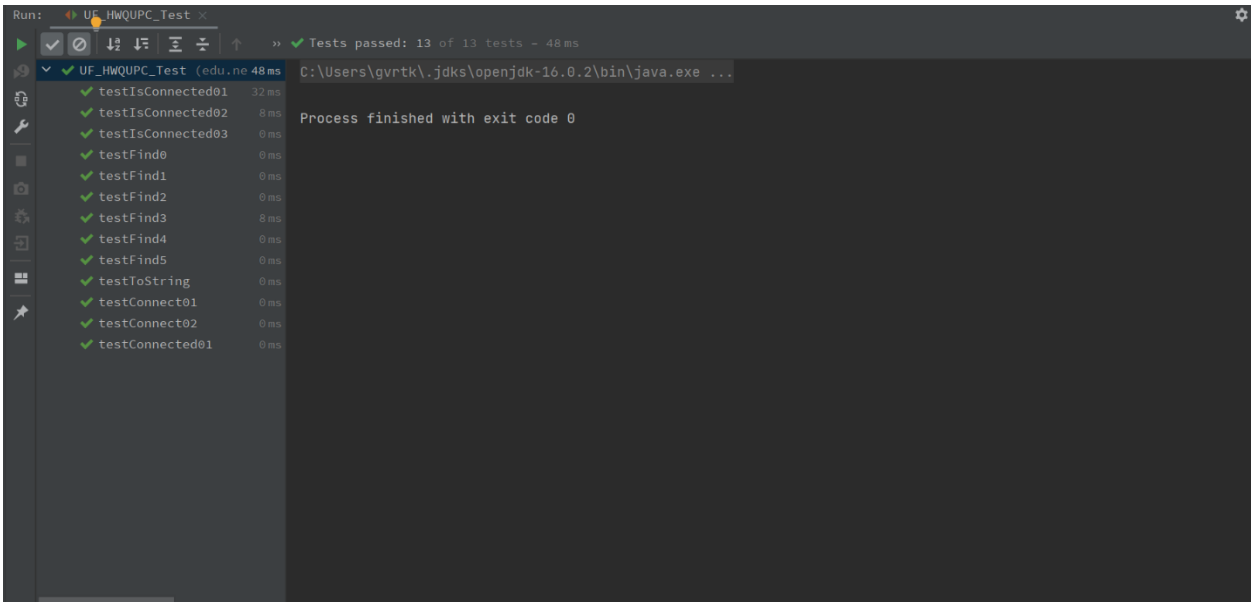
Assignment No. 3

Task 1

(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.

(b) Check that the unit tests for this class all work. You must show "green" test results in your submission (screenshot is OK).

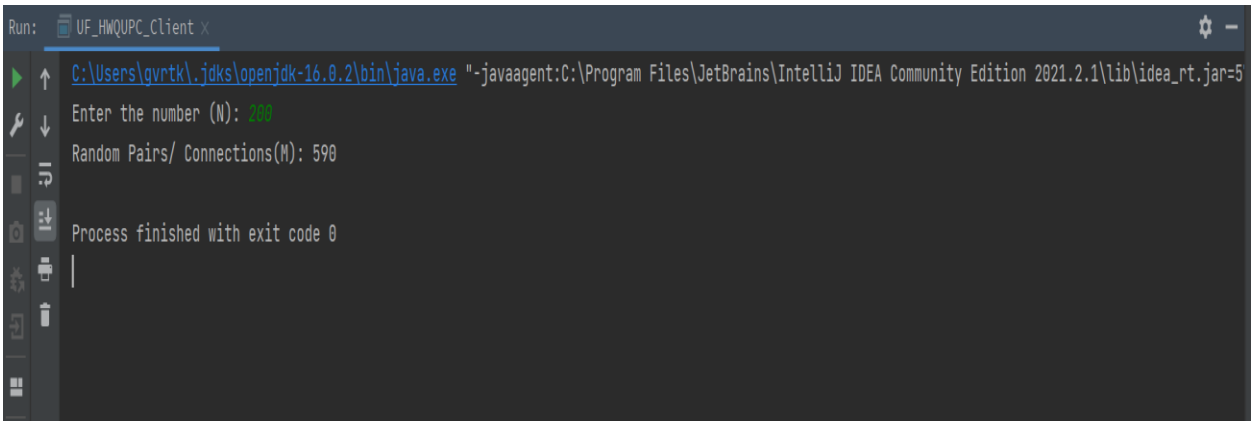
Output:



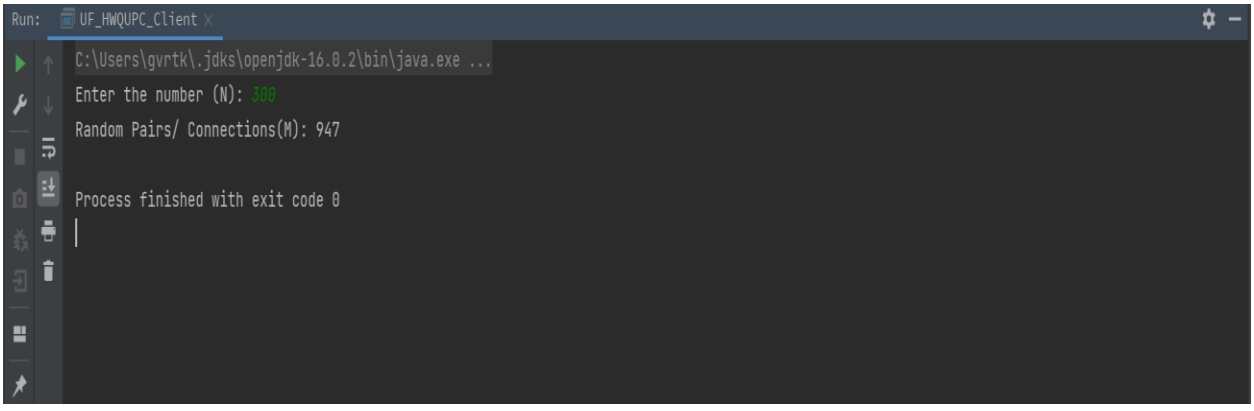
⊙ Task 2

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).

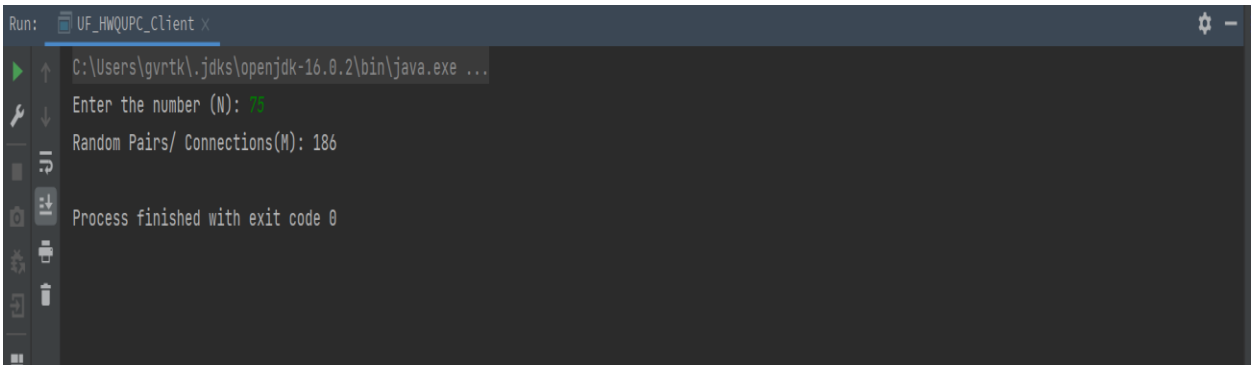
Output:



Test Result 1



Test Result 2



Test Result 3

⊙ 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:

Through many experiments , I have come to a conclusion that the number of generated pairs(**M**) for a given number of objects(**N**) is approximately **(N * ln(N))/2.**

$$M \approx (N*\ln(N))/2$$

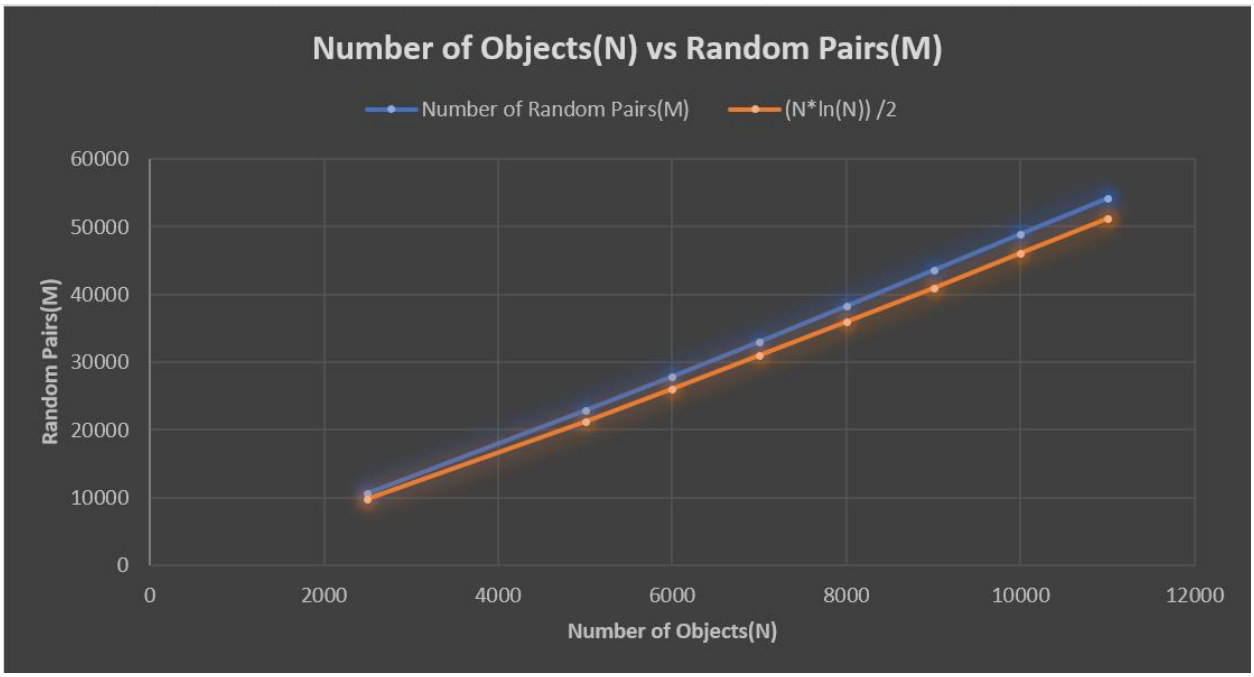
⊙ Evidence to support the conclusion:

Below, I have attached the results of the experiments I conducted and through data analysis through graph, we could see that the number of random pairs generated (M) is approximately equal to **(N*ln(N))/2)**

Here, we also see that the Weighted Quick-Union with Path Compression is somewhat linear though it is not so in theory.

Number (N)	Number of Random Pairs(M)	(N*ln(N)) /2
2500	10538	9780
5000	22790	21293
6000	27841	26099
7000	32965	30988
8000	38374	35949
9000	43657	40972
10000	48934	46052
11000	54273	51181

Experiment Data



Data Chart