XINHUI ZHAO (001560851)

Program Structures & Algorithms

Fall 2021

Assignment No. 3

Task (List down the tasks performed in the Assignment)

- Step 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).

- Step 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).

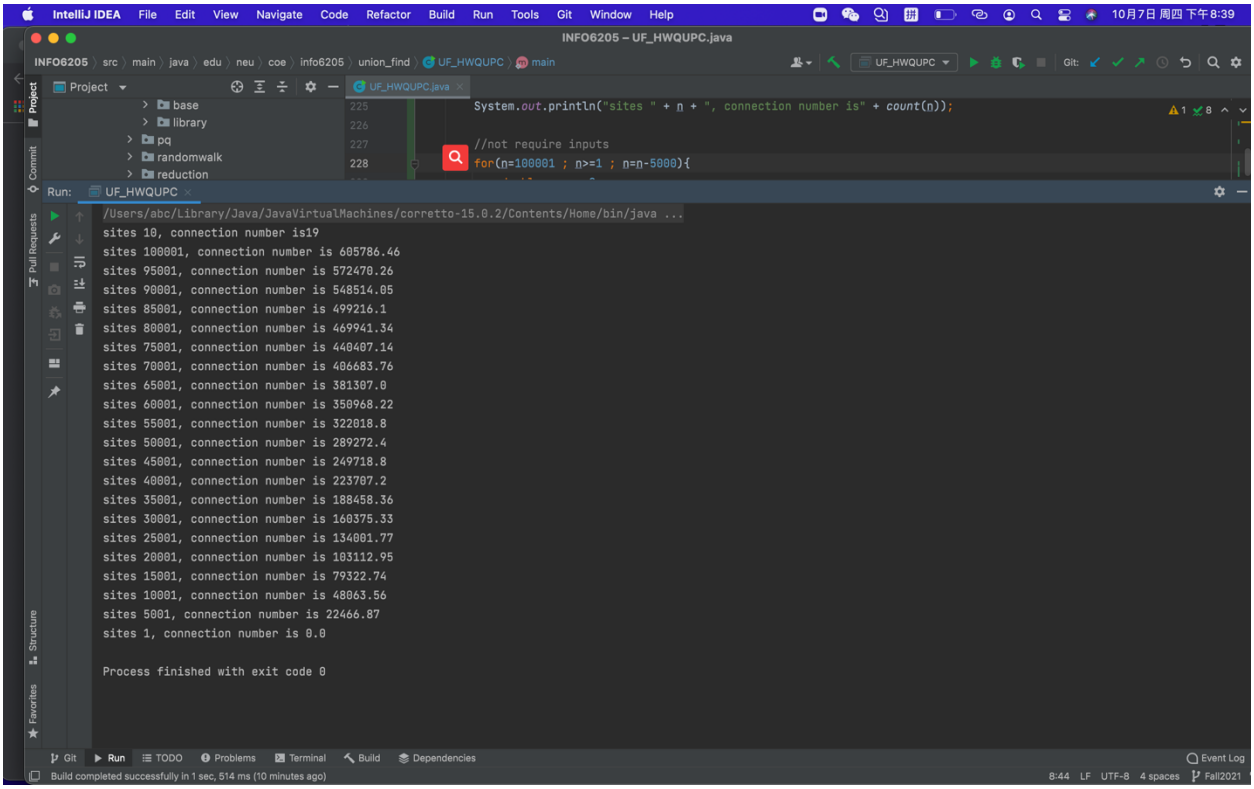
- Step 3:
  Determine the relationship between the number of objects (*n*) and the number of pairs (*m*) generated to accomplish this (i.e. to reduce the number of components from *n* to 1). Justify your conclusion in terms of your observations and what you think might be going on.

Relationship Conclusion: (For ex : z = a * b)

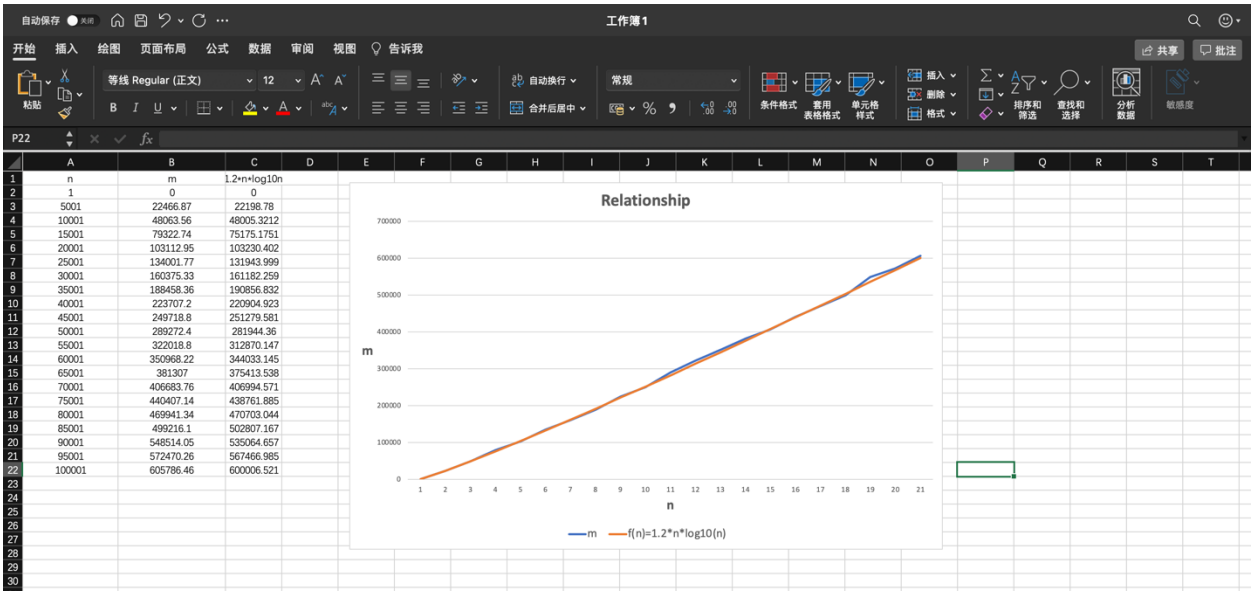
$$F(n)=1.2*n*\log10(n)$$

- **Evidence to support the conclusion:**

1. **Output (Snapshot of Code output in the terminal)**



2. **Graphical Representation(Observations from experiments should be tabulated and analyzed by plotting graphs(usually in excel) to arrive on the relationship conclusion)**



n	m	1.2*n*log10n
1	0	0
5001	22466.87	22198.78
10001	48063.56	48005.3212
15001	79322.74	75175.1751
20001	103112.95	103230.402
25001	134001.77	131943.999
30001	160375.33	161182.259
35001	188458.36	190856.832
40001	223707.2	220904.923
45001	249718.8	251279.581
50001	289272.4	281944.36
55001	322018.8	312870.147
60001	350968.22	344033.145
65001	381307	375413.538
70001	406683.76	406994.571
75001	440407.14	438761.885
80001	469941.34	470703.044
85001	499216.1	502807.167
90001	548514.05	535064.657
95001	572470.26	567466.985
100001	605786.46	600006.521

- **Unit tests result:(Snapshot of successful unit test run)**

Step1 unit test