Yang Song (001003647)

# Program Structures & Algorithms

# Fall 2021

# Assignment No. 3

- **Task (List down the tasks performed in the Assignment)**

  1. Implement height-weighted Quick Union with Path Compression
  2. 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.
  3. Determine the relationship between the number of objects (n) and the number of pairs (m)

- **Relationship Conclusion:**

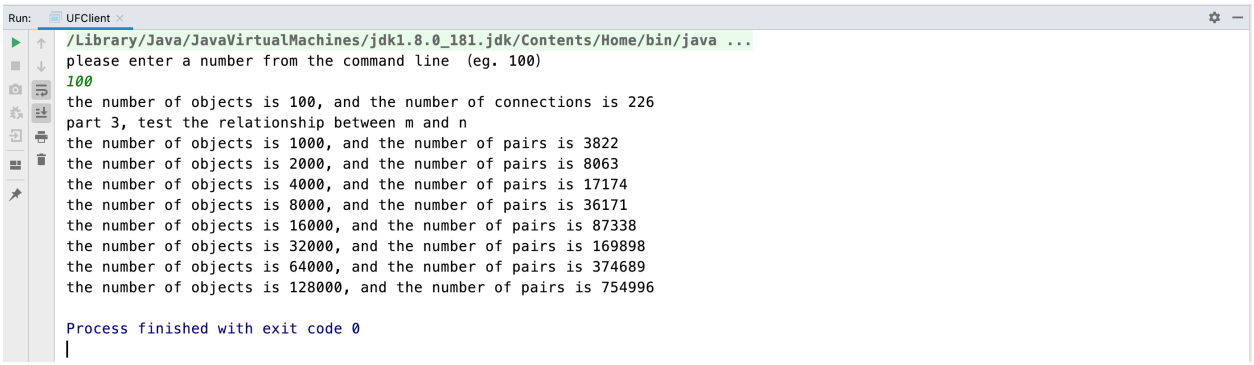
  $m = \frac{1}{2}nlnn$

- **Evidence to support the conclusion:**

1. **Output**

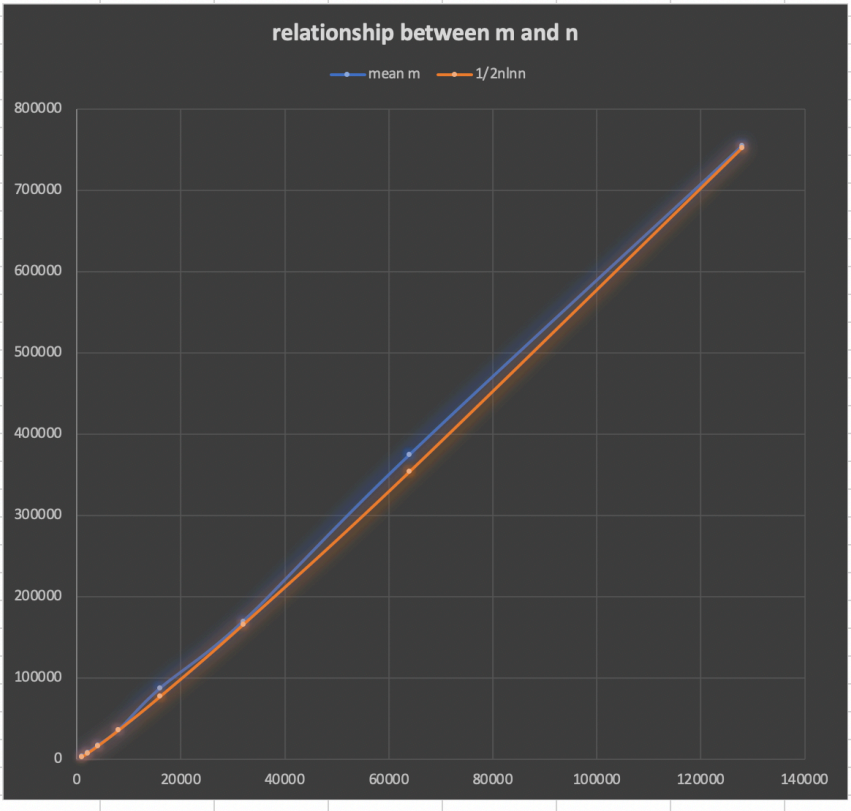
   Run main program in UFClient.java and. Firstly, we can input a number from the command line to test the count() method. Sec-

ondly, we can run more n values and make their values bigger using doubling method, each with 10 times to test the relationship between m and n.




## 2. **Graphical Representation**

| n | mean m | 1/2nlnn |
|---|---|---|
| 1000 | 3822 | 3454 |
| 2000 | 8063 | 7601 |
| 4000 | 17174 | 16588 |
| 8000 | 36171 | 35949 |
| 16000 | 87338 | 77443 |
| 32000 | 169898 | 165976 |
| 64000 | 374689 | 354132 |
| 128000 | 754996 | 752626 |




x-axis means n (the number of objects), blue line means 10 times mean random pairs of m and orange line means the function I assumed. We can find that two lines are similar, so m = $\frac{1}{2}$nlnn.

- **Unit tests result:**