

Program Structures and Algorithms
Spring 2023(SEC -03)

NAME:ALTAF SALIM SHAIKH
NUID:002774748

Task:

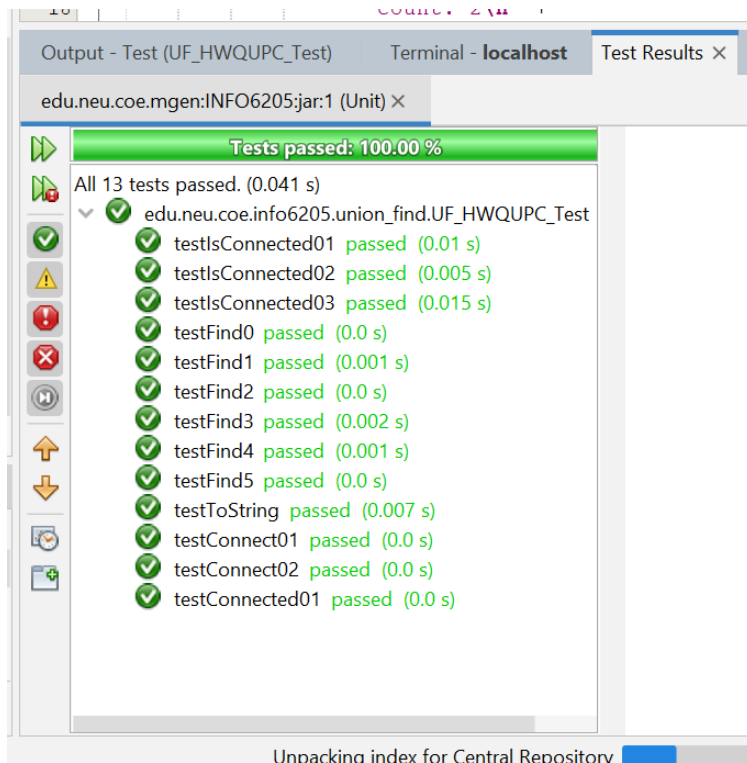
- Implement height-weighted Quick Union with Path Compression
- Develop a UF ("union-find") client that takes an integer value n from the command line to determine the number of "sites."
- Determine the relationship between the number of objects (n) and the number of pairs (m) generated

Step 1: UF_HWQUPC implementation snippets

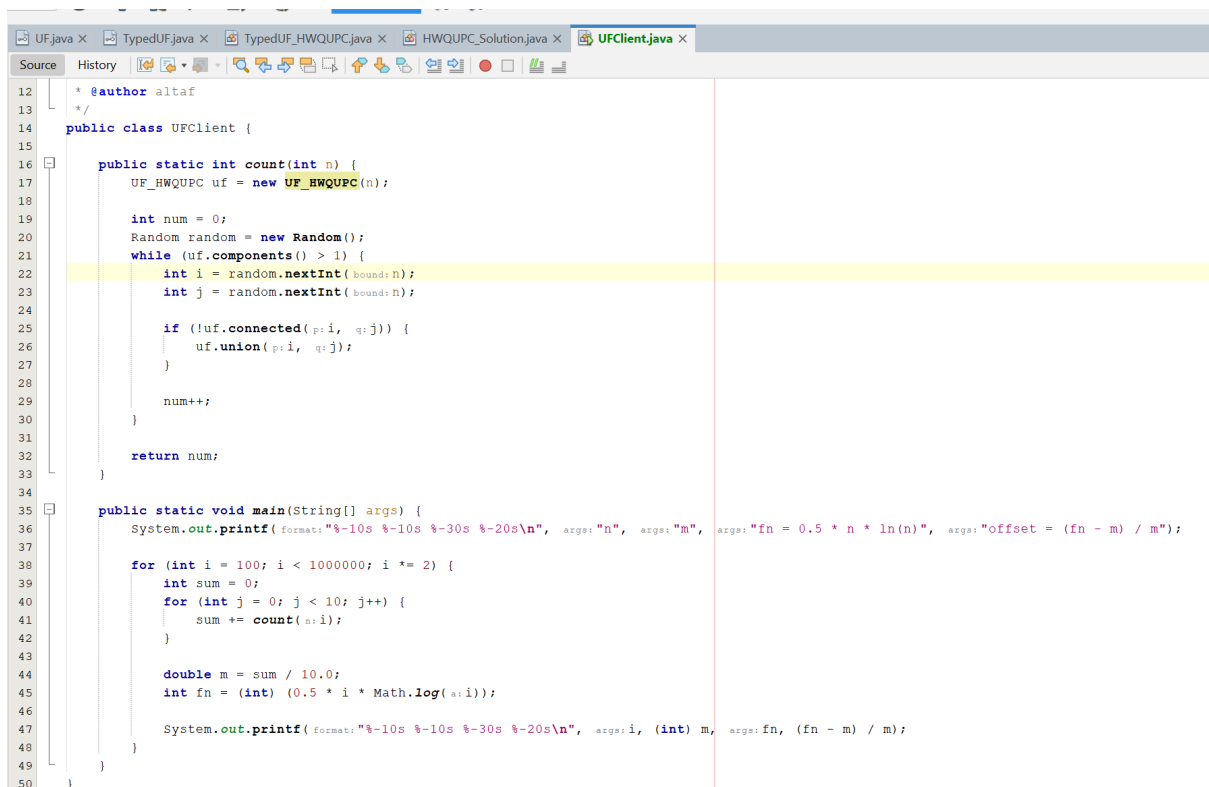
```
/** Returns the component identifier for the component containing site ...9 lines */
public int find(int p) {
    validate(p);
    int root = p;
    ////////////////////////////////////////
    // FIXME

    while (parent[root] != root) {
        doPathCompression(i:root);
        root = parent[root];
    }
    // END
    return root;
}
```

```
179
180 private void mergeComponents(int i, int j) {
181     ////////////////////////////////////////
182     // FIXME make shorter root point to taller one
183     if (height[j] > height[i]) {
184         parent[i] = j; height[j] += height[i]; height[i] = 0;
185     } else {
186         parent[j] = i; height[i] += height[j]; height[j] = 0;
187     }
188     // END
189 }
190 /**
191  * This implements the single-pass path-halving mechanism of path
192  * compression
193  */
194 private void doPathCompression(int i) {
195     // FIXME update parent to value of grandparent
196
197     ////////////////////////////////////////
198     if (this.pathCompression) {
199         parent[i] = parent[parent[i]];
200     }
201     // END
202 }
203 }
204
```



Step 2: UF Client implementation



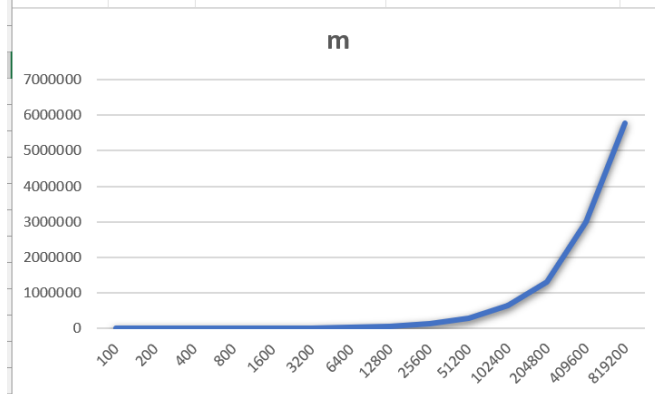
```

Output - Run (UFCClient) x Terminal - localhost Test Results
-----< edu.neu.coe.mgen:INFO6205 >-----
Building INFO6205 2023 1
-----[ jar ]-----
--- exec-maven-plugin:3.0.0:exec (default-cli) @ INFO6205 ---
n      m      fn = 0.5 * n * ln(n)      offset = (fn - m) / m
100    267    230      -0.14115011202389846
200    558    529      -0.05298961689939138
400    1421   1198     -0.15716898832137335
800    2923   2673     -0.08552856654122477
1600   6352   5902     -0.07096083616672966
3200   14554  12913    -0.11277079625952463
6400   28553  28044    -0.017840257202294697
12800  62406  60526    -0.03012530846392975
25600  139493  129924    -0.06859909199809887
51200  294126  277593    -0.05621124537698447
102400 645407  590676    -0.08480117903841504
204800 1316844 1252330    -0.04899180966379791
409600 2994866 2646617    -0.1162821447967714
819200 5770998 5577147    -0.0335905505425578

BUILD SUCCESS
-----
Total time: 6.522 s
Finished at: 2023-02-08T18:54:00-05:00
-----

```

n	m	fn = 0.5 * n * ln(n)	offset = (fn - m) / m
100	267	230	-0.141150112
200	558	529	-0.052989617
400	1421	1198	-0.157168988
800	2923	2673	-0.085528567
1600	6352	5902	-0.070960836
3200	14554	12913	-0.112770796
6400	28553	28044	-0.017840257
12800	62406	60526	-0.030125308
25600	139493	129924	-0.068599092
51200	294126	277593	-0.056211245
102400	645407	590676	-0.084801179
204800	1316844	1252330	-0.04899181
409600	2994866	2646617	-0.116282145
819200	5770998	5577147	-0.033590551



Relationship Conclusion:

The correlation between "n" and "m" can be estimated using the equation " $fn = 0.5 * n * \ln(n)$ ". "m" represents the number of connections needed to connect all the elements in a union find data structure, while "n" represents the number of elements in the data structure.

The relationship between "n" and "m" can be viewed as **logarithmic**, meaning that as the number of elements "n" increases, the number of connections "m" required also increases, but at a slower pace. The offset value shows how close the value of "m" is to the calculated value "fn", with a positive offset indicating that "m" is greater than "fn" and a negative offset indicating that "m" is less than "fn".