

Program Structures & Algorithms

Spring 2022

Assignment 3 (WQUPC)

Name: Aditya Kinare

(NUID): 001095881

- Task
- Output screenshot
- Relationship Conclusion
- Evidence / Graph
- Unit tests result

Task:

1. Implement height-weighted Quick Union with Path Compression.
2. Use WQUPC class to get multiple values of generated pairs for N components
3. Derive a relation between number of components(N), and number of Pairs(M) generated to create a cyclic graph.

Output screenshot:

Code Implementation for UF_HWQUPC

```
81• public int find(int p) {  
82     validate(p);  
83     int root = p;  
84     while(root != parent[root]) {  
85         if (this.pathCompression) {  
86             doPathCompression(root);  
87         }  
88         root = parent[root];  
89     }  
90     // END  
91     return root;  
92 }  
93
```

```

private void doPathCompression(int i) {
    // FIXME update parent to value of grandparent
    parent[i] = parent[parent[i]];
    // END
}

```

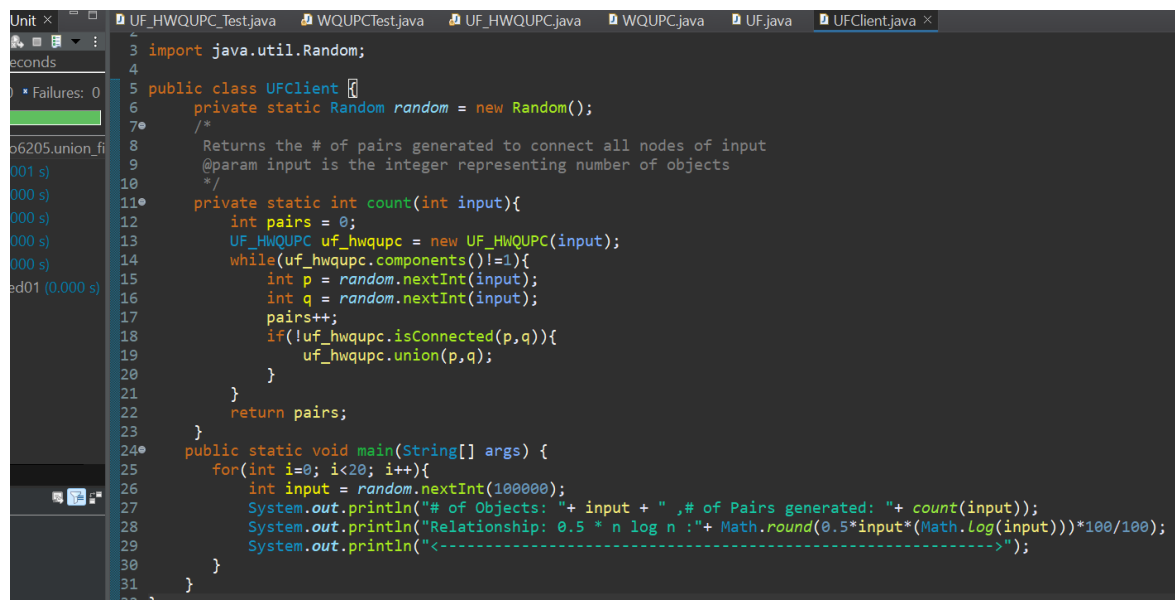
```

private void mergeComponents(int i, int j) {
    // FIXME make shorter root point to taller one
    int a = parent[i];
    int b = parent[j];

    if(height[a] < height[b]) {
        parent[a] = b;
        height[b] += height[a];
    }else{
        parent[b] = parent[a];
        height[a] += height [b];
    }
    // END
}

```

Implemented the UFClient.java class to get evidence for deriving a relation between N and M



```

Unit x
seconds
Failures: 0
p6205.union_fi
001 s)
000 s)
000 s)
000 s)
ed01 (0.000 s)

3 import java.util.Random;
4
5 public class UFClient {
6     private static Random random = new Random();
7     /*
8      Returns the # of pairs generated to connect all nodes of input
9      @param input is the integer representing number of objects
10     */
11     private static int count(int input){
12         int pairs = 0;
13         UF_HWQUPC uf_hwqupc = new UF_HWQUPC(input);
14         while(uf_hwqupc.components() != 1){
15             int p = random.nextInt(input);
16             int q = random.nextInt(input);
17             pairs++;
18             if(!uf_hwqupc.isConnected(p,q)){
19                 uf_hwqupc.union(p,q);
20             }
21         }
22         return pairs;
23     }
24     public static void main(String[] args) {
25         for(int i=0; i<20; i++){
26             int input = random.nextInt(100000);
27             System.out.println("# of Objects: "+ input + " ,# of Pairs generated: "+ count(input));
28             System.out.println("Relationship: 0.5 * n log n :"+ Math.round(0.5*input*(Math.Log(input)))*100/100);
29             System.out.println("<----->");
30         }
31     }
32 }

```

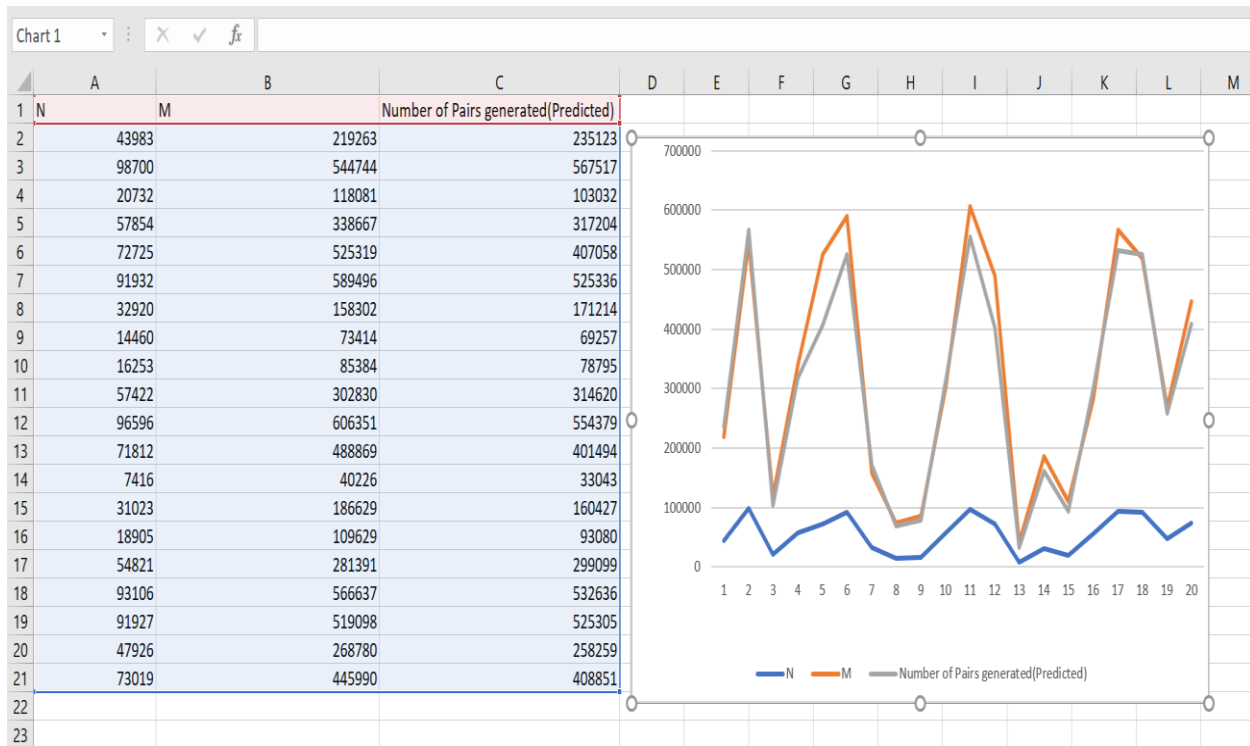
```
Problems Javadoc Declaration Console
<terminated> UFClient [Java Application] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (Mar 4, 2022, 8:35:23 PM - 8:35:23 PM)
# of Objects: 43983 ,# of Pairs generated: 219263
Relationship: 0.5 * n log n :235123
<----->
# of Objects: 98700 ,# of Pairs generated: 544744
Relationship: 0.5 * n log n :567517
<----->
# of Objects: 20732 ,# of Pairs generated: 118081
Relationship: 0.5 * n log n :103032
<----->
# of Objects: 57854 ,# of Pairs generated: 338667
Relationship: 0.5 * n log n :317204
<----->
# of Objects: 72725 ,# of Pairs generated: 525319
Relationship: 0.5 * n log n :407058
<----->
# of Objects: 91932 ,# of Pairs generated: 589496
Relationship: 0.5 * n log n :525336
<----->
# of Objects: 32920 ,# of Pairs generated: 158302
Relationship: 0.5 * n log n :171214
<----->
# of Objects: 14460 ,# of Pairs generated: 73414
Relationship: 0.5 * n log n :69257
<----->
# of Objects: 16253 ,# of Pairs generated: 85384
Relationship: 0.5 * n log n :78795
<----->
# of Objects: 57422 ,# of Pairs generated: 302830
Relationship: 0.5 * n log n :314620
```

Relationship Conclusion: The relationship between number of Objects (n) and number of pairs (m) generated to reach to single component i.e all objects should be connected is mentioned below:

Let n be number of objects and m be number of pairs generated to connect those objects

$$m = 0.5 * n \log n$$

Evidence/Graph:



Unit Test Results:

Package Explorer | JUnit

Finished after 0.041 seconds

Runs: 13/13 | Errors: 0 | Failures: 0

```

edu.neu.coe.info6205.union_find.UF_HWQUPC_Test [Runner: JUnit 4] (0.001 s)
  testIsConnected01 (0.000 s)
  testIsConnected02 (0.000 s)
  testIsConnected03 (0.000 s)
  testFind0 (0.000 s)
  testFind1 (0.001 s)
  testFind2 (0.000 s)
  testFind3 (0.000 s)
  testFind4 (0.000 s)
  testFind5 (0.000 s)
  testToString (0.000 s)
  testConnect01 (0.000 s)
  testConnect02 (0.000 s)
  testConnected01 (0.000 s)
  
```

```

154     assertEquals(0, h.find(1));
155     assertEquals(0, h.find(2));
156     assertEquals(0, h.find(3));
157     assertEquals(0, h.find(4));
158     assertEquals(0, h.find(5));
159     final PrivateMethodTester tester = new PrivateMethodTester(h);
160     assertEquals(0, tester.invokePrivate("getParent", 4));
161     assertEquals(0, tester.invokePrivate("getParent", 5));
162 }
163
164 /**
165  *
166  */
167 @Test(expected = IllegalArgumentException.class)
168 public void testFind5() {
169     UF h = new UF_HWQUPC(1);
170     h.find(1);
171 }
172
173 /**
174  *
175  */
176 /**
177 @Test
178 public void testConnected01() {
179     Connections h = new UF_HWQUPC(10);
180     h.show();
181     assertFalse(h.isConnected(0, 1));
182 }
183 }
  
```

Problems | Javadoc | Declaration | Console

<terminated> UF_HWQUPC_Test [JUnit] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (Mar 4, 2022, 8:31:26 PM - 8:31:26 PM)

