Program Structures and Algorithms
Spring 2024

NAME: Ashish Nevan Gade
NUID: 002889005
GITHUB LINK:

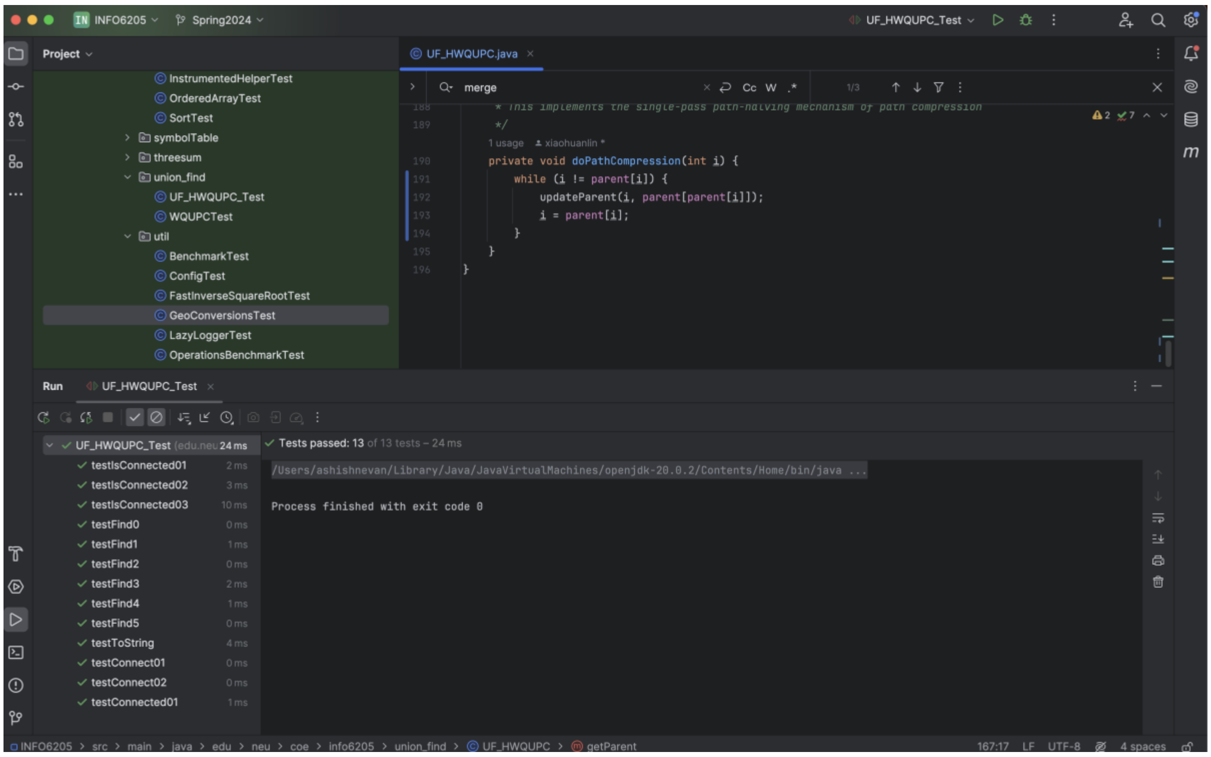
Task:

1. Implement height-weighted Quick Union with Path Compression
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.
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:

Evidence to support that conclusion:
the relationship between the number of objects (n) and the number of pairs (m) generated to connect all objects and reduce the number of components to 1 is linear. For 'n' objects, 'n-1' pairs are generated. This linear relationship is expected because each pair of objects (sites) forms one union operation, and 'n-1' such operations are required to connect all objects into one component.

Unit Test Screenshots:



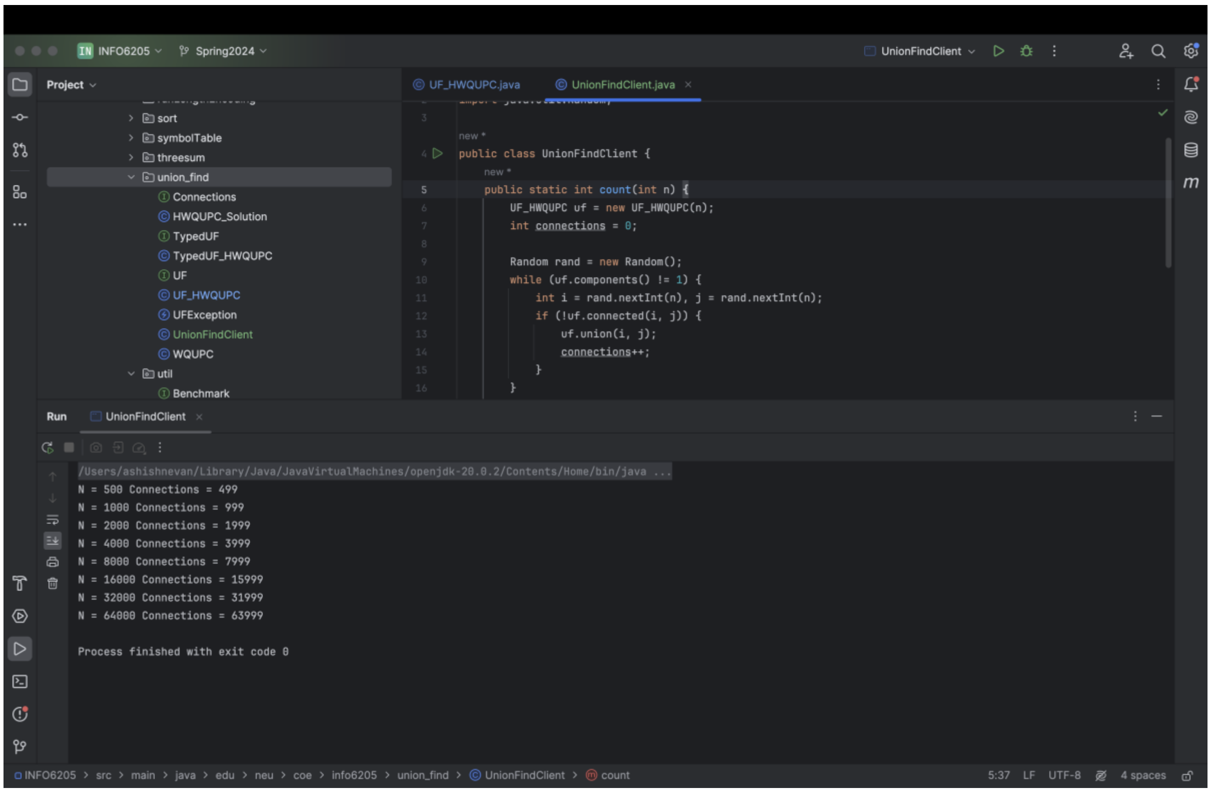
```java
     * This implements the single-pass path-halving mechanism of path compression
     */
    private void doPathCompression(int i) {
        while (i != parent[i]) {
            updateParent(i, parent[parent[i]]);
            i = parent[i];
        }
    }
}
```

```
UF_HWQUPC_Test (edu.neu 24 ms
  testIsConnected01   2 ms
  testIsConnected02   3 ms
  testIsConnected03  10 ms
  testFind0           0 ms
  testFind1           1 ms
  testFind2           0 ms
  testFind3           2 ms
  testFind4           1 ms
  testFind5           0 ms
  testToString        4 ms
  testConnect01       0 ms
  testConnect02       0 ms
  testConnected01     1 ms
```

```
Tests passed: 13 of 13 tests – 24 ms

/Users/ashishnevan/Library/Java/JavaVirtualMachines/openjdk-20.0.2/Contents/Home/bin/java ...

Process finished with exit code 0
```



```java
public class UnionFindClient {
    public static int count(int n) {
        UF_HWQUPC uf = new UF_HWQUPC(n);
        int connections = 0;

        Random rand = new Random();
        while (uf.components() != 1) {
            int i = rand.nextInt(n), j = rand.nextInt(n);
            if (!uf.connected(i, j)) {
                uf.union(i, j);
                connections++;
            }
        }
```

```
/Users/ashishnevan/Library/Java/JavaVirtualMachines/openjdk-20.0.2/Contents/Home/bin/java ...
N = 500 Connections = 499
N = 1000 Connections = 999
N = 2000 Connections = 1999
N = 4000 Connections = 3999
N = 8000 Connections = 7999
N = 16000 Connections = 15999
N = 32000 Connections = 31999
N = 64000 Connections = 63999

Process finished with exit code 0
```