Program Structures and Algorithms
Spring 2023(SEC 03) Assignment 04

NAME: Shivam Thabe
NUID: 002765286

Task: WQUPC

Part 1: Implement height-weighted Quick Union with Path Compression. For this, you will flesh out the class UF_HWQUPC. Check that the unit tests for this class all work. You must show "green" test results in your submission (screenshot is OK).

Part 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()

Part 3: Determine the relationship between the number of objects (*n*) and the number of pairs (*m*) generated to accomplish

Relationship Conclusion:

The relationship between m and n in the height-weighted quick-union with path compression algorithm is that m is proportional to n log(n).

This relationship is because the algorithm balances the height of the trees in the union-find data structure, which helps to reduce the time complexity of the operations. The weighted rule, which links the root of the smaller tree to the root of the larger tree, ensures that the height of the trees is logarithmic in the worst case.

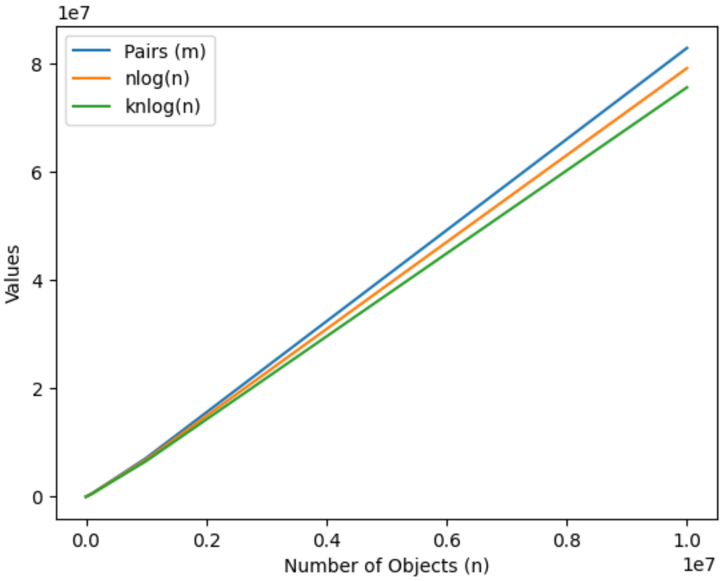
The path compression technique, which flattens the paths from the nodes to the roots, further reduces the height of the trees. The path compression technique adds another layer of optimization which reduces the complexity by a value of k, where k < 1.

These two optimizations combine to give the height-weighted quick-union with path compression algorithm a time complexity of k * n log(n), where k is less than 1.

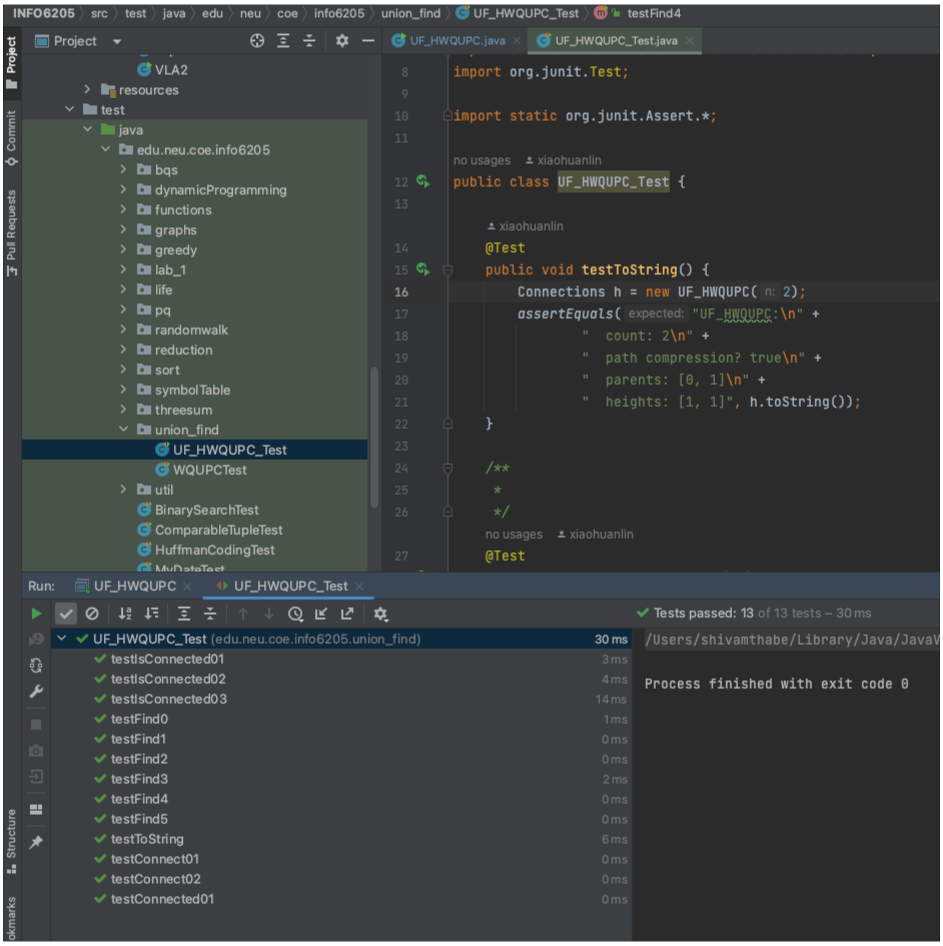
Observations:

No.	Total Objects (n)	Pairs (m)	nlogn	k
1	10	16	27.72588722	0.577078016
2	100	260	556.0681631	0.467568577
3	1000	3758	8231.64218	0.45653102
4	10000	47900	107768.7078	0.444470394
5	100000	606622	1331566.114	0.455570319
6	1000000	7137121	15780820.03	0.452265534
7	10000000	82908866	182332525.6	0.454712431

Graphical Representation:



Unit Test Screenshots:



```
import org.junit.Test;

import static org.junit.Assert.*;

no usages  xiaohuanlin
public class UF_HWQUPC_Test {

     xiaohuanlin
    @Test
    public void testToString() {
        Connections h = new UF_HWQUPC( n: 2);
        assertEquals( expected: "UF_HWQUPC:\n" +
                "  count: 2\n" +
                "  path compression? true\n" +
                "  parents: [0, 1]\n" +
                "  heights: [1, 1]", h.toString());
    }

    /**
     *
     */
    no usages  xiaohuanlin
    @Test
```

```
Tests passed: 13 of 13 tests – 30 ms

UF_HWQUPC_Test (edu.neu.coe.info6205.union_find)   30 ms
    testIsConnected01    3 ms
    testIsConnected02    4 ms
    testIsConnected03   14 ms
    testFind0            1 ms
    testFind1            0 ms
    testFind2            0 ms
    testFind3            2 ms
    testFind4            0 ms
    testFind5            0 ms
    testToString         6 ms
    testConnect01        0 ms
    testConnect02        0 ms
    testConnected01      0 ms
```

```
/Users/shivamthabe/Library/Java/JavaV

Process finished with exit code 0
```