

Program Structures and Algorithms

Spring 2024

NAME: Zenan Fan

NUID: 002854067

GITHUB LINK: <https://github.com/LearningMachine/INFO6205>

Task: Assignment 4 (WQUPC)

Unit Test Screenshots:

```
public class UF_HWQUPC_Test {  
    // xiaohuanlin  
    @Test  
    public void testToString() {  
        Connections h = new UF_HWQUPC(10, 2);  
        assertEquals("UF_HWQUPC:\n" +  
            "  count: 2\n" +  
            "  path compression? true\n" +  
            "  parents: [0, 1]\n" +  
            "  heights: [1, 1]", h.toString());  
    }  
    /**  
     *  
     */  
    // xiaohuanlin  
    @Test  
    public void testIsConnected01() {  
        Connections h = new UF_HWQUPC(10, 2);  
        assertFalse(h.isConnected(0, 1));  
    }  
}
```

Run UF_HWQUPC_Test x

✓ UF_HWQUPC_Test (edu.neu.coe.info6205) 16 ms ✓ Tests passed: 13 of 13 tests - 16 ms
/Library/Java/JavaVirtualMachines/jdk-17-jdk/Contents/Home/bin/java ...
Process finished with exit code 0

FO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test > testFind2

Implementation of UF_HWQUPC:

```
public class UF_Implementation {  
    public static int count(int n) {  
        int connection = 0;  
        Random random = new Random();  
        UF_HWQUPC uf_hwqupc = new UF_HWQUPC(n);  
        int loopTime = n;  
        while (loopTime > 1) {  
            int p = random.nextInt(n);  
            int q = random.nextInt(n);  
            connection++;  
            if (!uf_hwqupc.isConnected(p, q)) {  
                uf_hwqupc.union(p, q);  
            }  
            loopTime--;  
        }  
        return connection;  
    }  
}
```

Run Run x

/Library/Java/JavaVirtualMachines/jdk-17-jdk/Contents/Home/bin/java ...
Length = 50, number of pairs generated for the UF: 124
Length = 100, number of pairs generated for the UF: 277
Length = 200, number of pairs generated for the UF: 579
Length = 400, number of pairs generated for the UF: 1250
Length = 800, number of pairs generated for the UF: 2974
Length = 1600, number of pairs generated for the UF: 6650
Process finished with exit code 0

INFO6205 > src > main > java > edu > neu > coe > info6205 > union_find > UF_Implementation.java > Run > main

Relationship Conclusion: There is a super linear relationship between m and n , but with a lower growth rate than the quadratic, $y = 1.15x + 0.23$, in which y is $\log(m)$, x is $\log(n)$