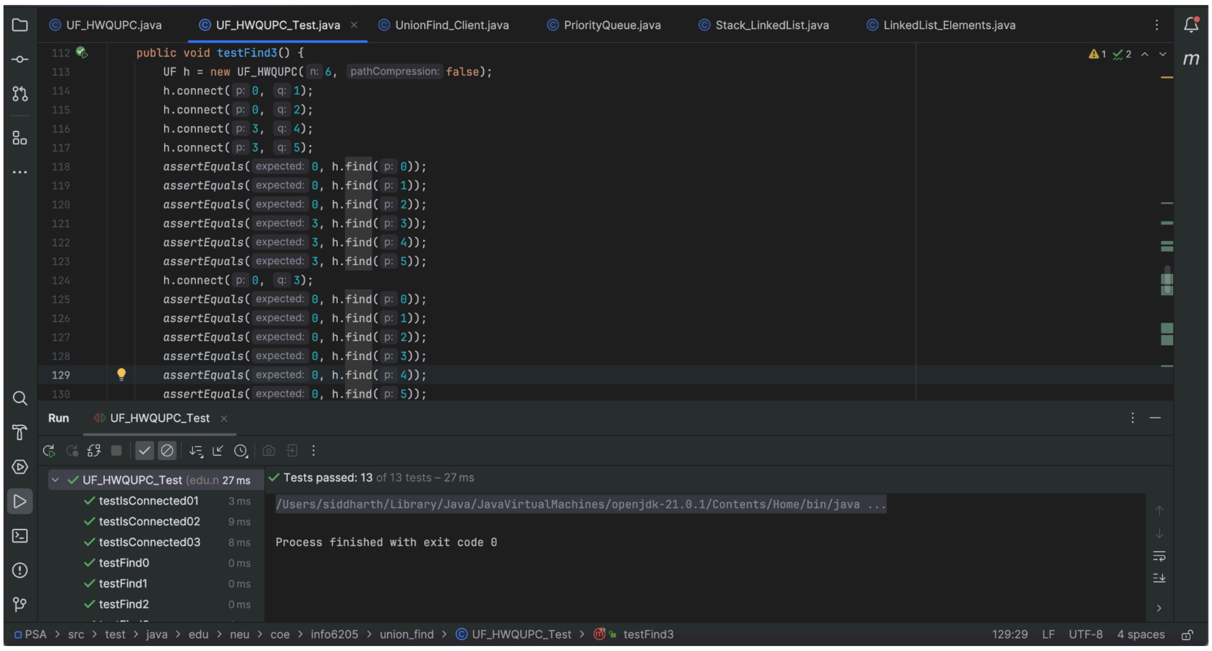
Program Structures and Algorithms

Spring 2024

NAME: Siddharth Dumbre
NUID: 002247119
GITHUB LINK: https://github.com/dumbresi/Info-6205-spring2024

Task: WQUPC- Weighted Quick Union Path Compression

Output screenshot:



```java
    public void testFind3() {
        UF h = new UF_HWQUPC( n: 6,  pathCompression: false);
        h.connect( p: 0,  q: 1);
        h.connect( p: 0,  q: 2);
        h.connect( p: 3,  q: 4);
        h.connect( p: 3,  q: 5);
        assertEquals( expected: 0, h.find( p: 0));
        assertEquals( expected: 0, h.find( p: 1));
        assertEquals( expected: 0, h.find( p: 2));
        assertEquals( expected: 3, h.find( p: 3));
        assertEquals( expected: 3, h.find( p: 4));
        assertEquals( expected: 3, h.find( p: 5));
        h.connect( p: 0,  q: 3);
        assertEquals( expected: 0, h.find( p: 0));
        assertEquals( expected: 0, h.find( p: 1));
        assertEquals( expected: 0, h.find( p: 2));
        assertEquals( expected: 0, h.find( p: 3));
        assertEquals( expected: 0, h.find( p: 4));
        assertEquals( expected: 0, h.find( p: 5));
```

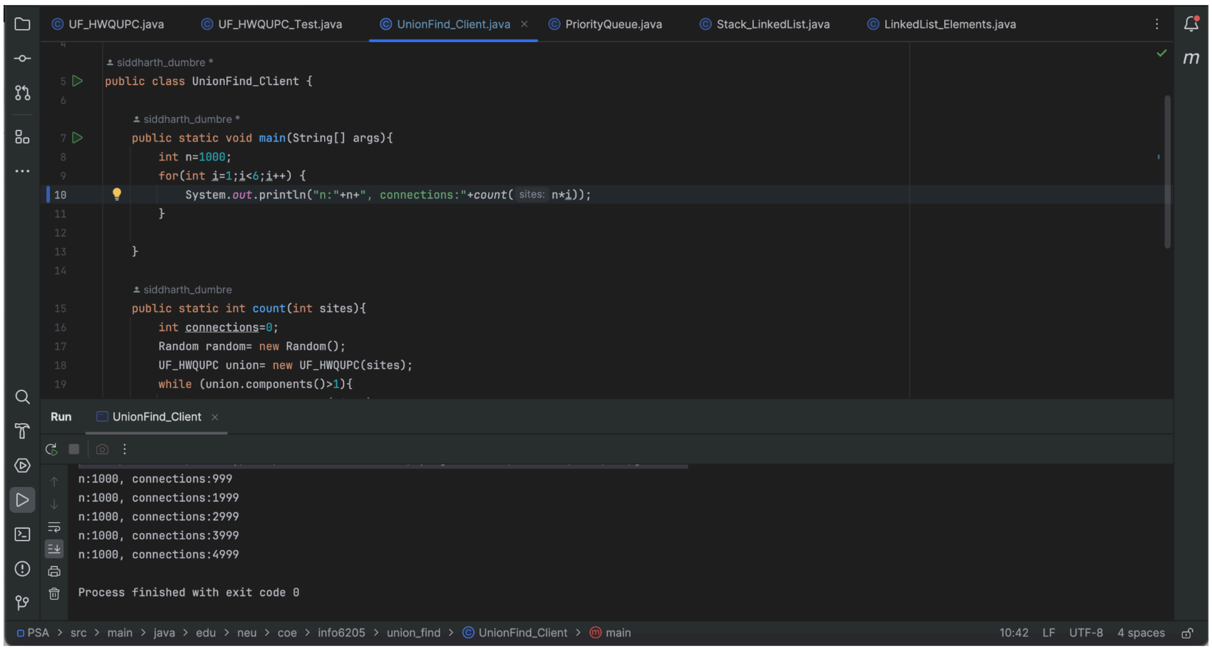
```
✓ UF_HWQUPC_Test (edu.n 27 ms
    ✓ testIsConnected01   3 ms
    ✓ testIsConnected02   9 ms
    ✓ testIsConnected03   8 ms
    ✓ testFind0           0 ms
    ✓ testFind1           0 ms
    ✓ testFind2           0 ms
```

```
✓ Tests passed: 13 of 13 tests – 27 ms

/Users/siddharth/Library/Java/JavaVirtualMachines/openjdk-21.0.1/Contents/Home/bin/java ...

Process finished with exit code 0
```

PSA > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test > testFind3
129:29 LF UTF-8 4 spaces



```java
siddharth_dumbre *
public class UnionFind_Client {

    siddharth_dumbre *
    public static void main(String[] args){
        int n=1000;
        for(int i=1;i<6;i++) {
            System.out.println("n:"+n+", connections:"+count( sites: n*i));
        }

    }

    siddharth_dumbre
    public static int count(int sites){
        int connections=0;
        Random random= new Random();
        UF_HWQUPC union= new UF_HWQUPC(sites);
        while (union.components()>1){
```

```
n:1000, connections:999
n:1000, connections:1999
n:1000, connections:2999
n:1000, connections:3999
n:1000, connections:4999

Process finished with exit code 0
```

Observation:

The relationship between the number of objects(n) and the number of pairs is that the number of pairs(m) is 1 less than the number of sites. I.e. m=n-1.