

Assignment 4 Union Find

Part 1

Screenshots of running Unit Tests

```
INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test.java
Project Commit Pull Requests Bookmarks Structure
UF_HWQUPC.java × UF_HWQUPC_Test.java ×
no usages  ↳ xiaohuanlin
12  ↳ public class UF_HWQUPC_Test {
13
    ↳     ↳ xiaohuanlin
14    ↳     @Test
15    ↳     public void testToString() {
16        ↳         Connections h = new UF_HWQUPC(n: 2);
17        ↳         assertEquals(expected: "UF_HWQUPC:\n" +
18        ↳             " count: 2\n" +
19        ↳             " path compression? true\n" +
20        ↳             " parents: [0, 1]\n" +
21        ↳             " heights: [1, 1]", h.toString());
22    }
23
24    ↳     /**
25    ↳     *
26    ↳     */
27    ↳     no usages  ↳ xiaohuanlin
28    ↳     @Test
29    ↳     public void testIsConnected01() {
30        ↳         Connections h = new UF_HWQUPC(n: 2);
31        ↳         assertFalse(h.isConnected(p: 0, q: 1));
32
33    ↳     /**
34    ↳     *
35    ↳     */
36    ↳     no usages  ↳ xiaohuanlin
37    ↳     @Test(expected = IllegalArgumentException.class)
38    ↳     public void testIsConnected02() {
39        ↳         Connections h = new UF_HWQUPC(n: 1);
40        ↳         assertTrue(h.isConnected(p: 0, q: 1));
41
42    ↳     /**
43    ↳     *
Tests passed: 3 (moments ago)
```

INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF

UF_HWQUPC.java < UF_HWQUPC_Test.java <

```
45 @Test
46     public void testIsConnected03() {
47         Connections h = new UF_HWQUPC(n: 2);
48         final PrivateMethodTester tester = new PrivateMethodTester(h);
49         assertNull(tester.invokePrivate("upc", "isConnected", "0", "1"));
50         assertTrue(h.isConnected(p: 0, q: 1));
51     }
52
53     /**
54      *
55      */
56     no usages ↗ xiaohuanlin
57 @Test
58     public void testConnect01() {
59         Connections h = new UF_HWQUPC(n: 2);
60         h.connect(p: 0, q: 1);
61     }
62
63     /**
64      *
65      */
66     no usages ↗ xiaohuanlin
67 @Test
68     public void testConnect02() {
69         Connections h = new UF_HWQUPC(n: 2);
70         h.connect(p: 0, q: 1);
71         h.connect(p: 0, q: 1);
72         assertTrue(h.isConnected(p: 0, q: 1));
73     }
74
75     /**
76      *
77     no usages ↗ xiaohuanlin
78 @Test
79     public void testFind0() {
80         IIF h = new IIF_HWQUPC(n: 1);
81     }

```

Git Run TODO Problems Terminal Services Build

Tests passed: 13 (2 minutes ago)

INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test

UF_HWQUPC.java < UF_HWQUPC_Test.java

Project Commit Pull Requests Bookmarks Structure

no usages xiaohuanlin

```
85     @Test
86     public void testFind1() {
87         UF h = new UF_HWQUPC(n: 2);
88         h.connect(p: 0, q: 1);
89         assertEquals(expected: 0, h.find(p: 0));
90         assertEquals(expected: 0, h.find(p: 1));
91     }
92
93     /**
94      *
95     */
96     no usages xiaohuanlin
97     @Test
98     public void testFind2() {
99         UF h = new UF_HWQUPC(n: 3, pathCompression: false);
100        h.connect(p: 0, q: 1);
101        assertEquals(expected: 0, h.find(p: 0));
102        assertEquals(expected: 0, h.find(p: 1));
103        h.connect(p: 2, q: 1);
104        assertEquals(expected: 0, h.find(p: 0));
105        assertEquals(expected: 0, h.find(p: 1));
106        assertEquals(expected: 0, h.find(p: 2));
107    }
108    /**
109     *
110    */
111    no usages xiaohuanlin
112    @Test
113    public void testFind3() {
114        UF h = new UF_HWQUPC(n: 6, pathCompression: false);
115        h.connect(p: 0, q: 1);
116        h.connect(p: 0, q: 2);
117        h.connect(p: 3, q: 4);
118        h.connect(p: 3, q: 5);
119        assertEquals(expected: 0, h.find(p: 0));
120    }

```

Git Run TODO Problems Terminal Services Build Dependencies

Tests passed: 13 (3 minutes ago)

INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test

UF_HWQUPC.java × UF_HWQUPC_Test.java ×

no usages ▾ xiaohuanlin

```
139
140     @Test
141     public void testFind4() {
142         UF h = new UF_HWQUPC( n: 6);
143         h.connect( p: 0, q: 1);
144         h.connect( p: 0, q: 2);
145         h.connect( p: 3, q: 4);
146         h.connect( p: 3, q: 5);
147         assertEquals( expected: 0, h.find( p: 0));
148         assertEquals( expected: 0, h.find( p: 1));
149         assertEquals( expected: 0, h.find( p: 2));
150         assertEquals( expected: 3, h.find( p: 3));
151         assertEquals( expected: 3, h.find( p: 4));
152         assertEquals( expected: 3, h.find( p: 5));
153         h.connect( p: 0, q: 3);
154         assertEquals( expected: 0, h.find( p: 0));
155         assertEquals( expected: 0, h.find( p: 1));
156         assertEquals( expected: 0, h.find( p: 2));
157         assertEquals( expected: 0, h.find( p: 3));
158         assertEquals( expected: 0, h.find( p: 4));
159         assertEquals( expected: 0, h.find( p: 5));
160         final PrivateMethodTester tester = new PrivateMethodTester(h);
161         assertEquals( expected: 0, tester.invokePrivate( name: "getParent", ...parameters: 4));
162         assertEquals( expected: 0, tester.invokePrivate( name: "getParent", ...parameters: 5));
163
164     /**
165      *
166      */
167     no usages ▾ xiaohuanlin
168     @Test(expected = IllegalArgumentException.class)
169     public void testFind5() {
```

Project Commit Pull Requests Bookmarks Structure Git Run TODO Problems Terminal Services Build Dependencies

INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC_Test > testFind4

UF_HWQUPC.java > UF_HWQUPC_Test.java

```
139
140     @Test
141     public void testFind4() {
142         UF h = new UF_HWQUPC( n: 6 );
143         h.connect( p: 0, q: 1 );
144         h.connect( p: 0, q: 2 );
145         h.connect( p: 3, q: 4 );
146         h.connect( p: 3, q: 5 );
147         assertEquals( expected: 0, h.find( p: 0 ) );
148         assertEquals( expected: 0, h.find( p: 1 ) );
149         assertEquals( expected: 0, h.find( p: 2 ) );
150         assertEquals( expected: 3, h.find( p: 3 ) );
151         assertEquals( expected: 3, h.find( p: 4 ) );
152         assertEquals( expected: 3, h.find( p: 5 ) );
153         h.connect( p: 0, q: 3 );
154         assertEquals( expected: 0, h.find( p: 0 ) );
155         assertEquals( expected: 0, h.find( p: 1 ) );
156         assertEquals( expected: 0, h.find( p: 2 ) );
157         assertEquals( expected: 0, h.find( p: 3 ) );
158         assertEquals( expected: 0, h.find( p: 4 ) );
159         assertEquals( expected: 0, h.find( p: 5 ) );
160         final PrivateMethodTester tester = new PrivateMethodTester(h);
161         assertEquals( expected: 0, tester.invokePrivate( name: "getParent", ...parameters: 4 ) );
162         assertEquals( expected: 0, tester.invokePrivate( name: "getParent", ...parameters: 5 ) );
163     }
164     /**
165      *
166      */
167     no usages  ↳ xiaohuanlin
168     @Test(expected = IllegalArgumentException.class)
169     public void testFind5() {
170         UF h = new UF_HWQUPC( n: 1 );
171         h.find( p: 1 );
172     }
173     /**
174      */

Tests passed: 13 (5 minutes ago)
```

```
INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC.java > UF_HWQUPC_Test.java > testFind5

Project Commit Pull Requests Bookmarks
INFO6205 > src > test > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC.java > UF_HWQUPC_Test.java > testFind5

159     final PrivateMethodTester tester = new PrivateMethodTester(h);
160     assertEquals( expected: 0, tester.invokePrivate( name: "getParent",
161     assertEquals( expected: 0, tester.invokePrivate( name: "getParent",
162     }
163
164     /**
165     *
166     */
167     no usages  ↳ xiaohuanlin
168     @Test(expected = IllegalArgumentException.class)
169     public void testFind5() {
170         UF h = new UF_HWQUPC( n: 1);
171         h.find( p: 1);
172
173
174     /**
175     *
176     */
177     no usages  ↳ xiaohuanlin
178     @Test
179     public void testConnected01() {
180         Connections h = new UF_HWQUPC( n: 10);
181         h.show();
182         assertFalse(h.isConnected( p: 0, q: 1));
183     }
```

Part 2 Program Execution

The screenshot shows a Java IDE interface with the following details:

- Project Bar:** INFO6205 / src / main / java / edu / neu / coe / info6205 / union_find / UF_HWQUPC / UF_HWQUPC
- Run Bar:** UF_HWQUPC.java, UF_HWQUPC_Test.java, Run: UF_HWQUPC
- Output Window:** Displays the execution of the UF_HWQUPC program for different site counts (100, 200, 400, 800, 1600). The output includes:
 - Client starts
 - Graph is all connected
 - Number of connect(a,b) function calls
 - Number of valid connect(a,b) function calls
 - Error message: "(connect(a,b) is not valid if a b is already connected before the function is called)"
 - Client ends
- Bottom Status Bar:** Git, Run, TODO, Problems, Terminal, Services, Build, Dependencies
- Bottom Message:** All files are up-to-date (moments ago)

INFO6205 > src > main > java > edu > neu > coe > info6205 > union_find > UF_HWQUPC > UF_HWQUPC

UF_HWQUPC.java < UF_HWQUPC_Test.java <

Run: UF_HWQUPC

(connect(a,b) is not valid if a b is already connected before the function is called)
Client ends

Client with 1600 sites starts
Graph is all connected
num of connect(a,b) function called 7126 times
num of of valid connect(a,b) function called 1599 times
(connect(a,b) is not valid if a b is already connected before the function is called)
Client ends

Client with 3200 sites starts
Graph is all connected
num of connect(a,b) function called 14271 times
num of of valid connect(a,b) function called 3199 times
(connect(a,b) is not valid if a b is already connected before the function is called)
Client ends

Client with 6400 sites starts
Graph is all connected
num of connect(a,b) function called 25713 times
num of of valid connect(a,b) function called 6399 times
(connect(a,b) is not valid if a b is already connected before the function is called)
Client ends

Client with 12800 sites starts
Graph is all connected
num of connect(a,b) function called 67134 times
num of of valid connect(a,b) function called 12799 times
(connect(a,b) is not valid if a b is already connected before the function is called)
Client ends

All files are up-to-date (a minute ago)

Part 3 Observations

N(Number of sites)	M (Number of connections)
100	99
200	199
400	399
800	799
1600	1599
3200	3199
6400	6399

Relationship between n (number of sites) and m (number of connections) is
 $m=n-1$

