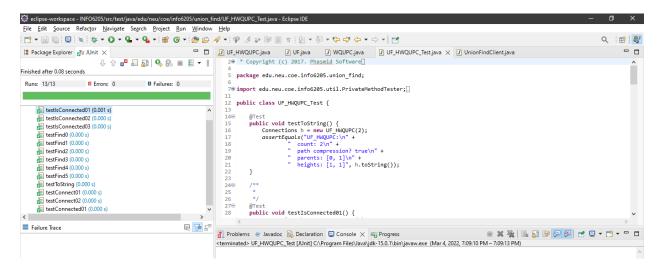
Step: 1 Implementation of height-weighted Quick Union with Path Compression.

Test Case Screenshot:



<u>Step 2</u>: Implementation of UF_HWQUPC, develop a UnionFindClient file.

Result:

```
The problems @ Javadoc Declaration Console X progress

**Cerminated DinonFindClient [Java Application] Console X progress

**Cerminated DinonFindClient [Java Application] Console X progress

**Cerminated DinonFindClient [Java Application] Console X progress

**Integer value: 100, number of sites: 263.88 expected value 244.07401985736885 ratio 1.0811474328738688

**Integer value: 200, number of sites: 611.7 expected value 561.6216408540919 ratio 1.089167314218929

**Integer value: 300, number of sites: 15.22 expected value 966.90411347493359 ratio 1.0091725367345414

**Integer value: 400, number of sites: 1363.05 expected value 1270.190483986892 ratio 1.0731067640513563

**Integer value: 500, number of sites: 1717.76 expected value 1646.8711460818806 ratio 1.0430445661075387

**Integer value: 600, number of sites: 1717.66 expected value 2034.2236393867345 ratio 1.04010613211143857

**Integer value: 700, number of sites: 2463.12 expected value 2430.450804301103 ratio 1.0134416198184646

**Integer value: 800, number of sites: 2935.13 expected value 2834.275372531201 ratio 1.0355839197723153

**Integer value: 900, number of sites: 3836.64 expected value 2844.7743205096 ratio 1.0252401239510318

**Integer value: 1000, number of sites: 3806.16 expected value 3661.1102978605327 ratio 1.0396190473213087
```

Conclusion:

The relationship between number of objects(n) and number of pairs(m): m = 0.53* n* lg(n)

Below table and graph shows relationship between number of pairs and expected value. It depicts that the relationship between number of pairs and expected values is almost 1.

Number of pairs	Expected Value
263.88	244.074
611.7	561.621
915.22	906.901
1363.05	1270.19
1717.76	1646.871
2117.66	2034.223
2463.12	2430.45
2935.13	2834.275
3326.64	3244.742
3806.16	3661.11

