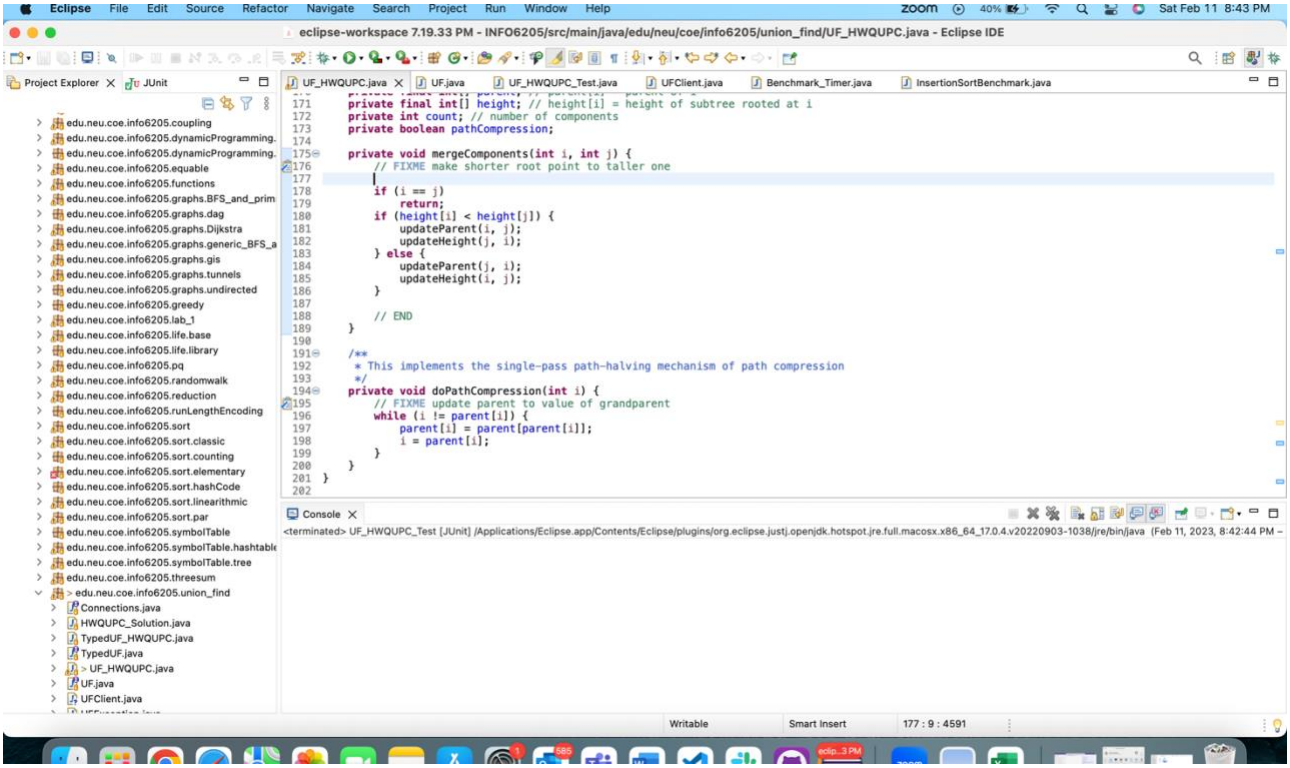
# Program Structures and Algorithms Sec -8

## **Name:** Uday Kiran Reddy Mulpuri **NUID:** 002781063

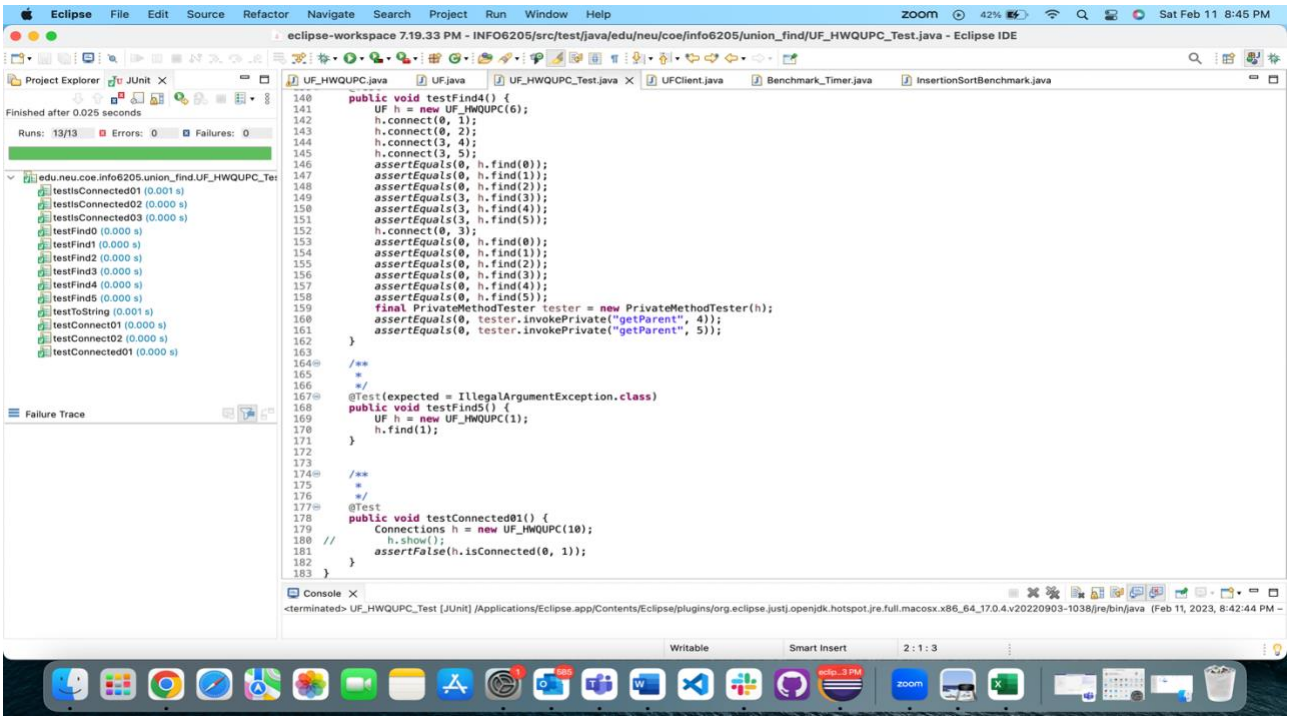
**Assignment-4(WQUPC)**

## Step 1:

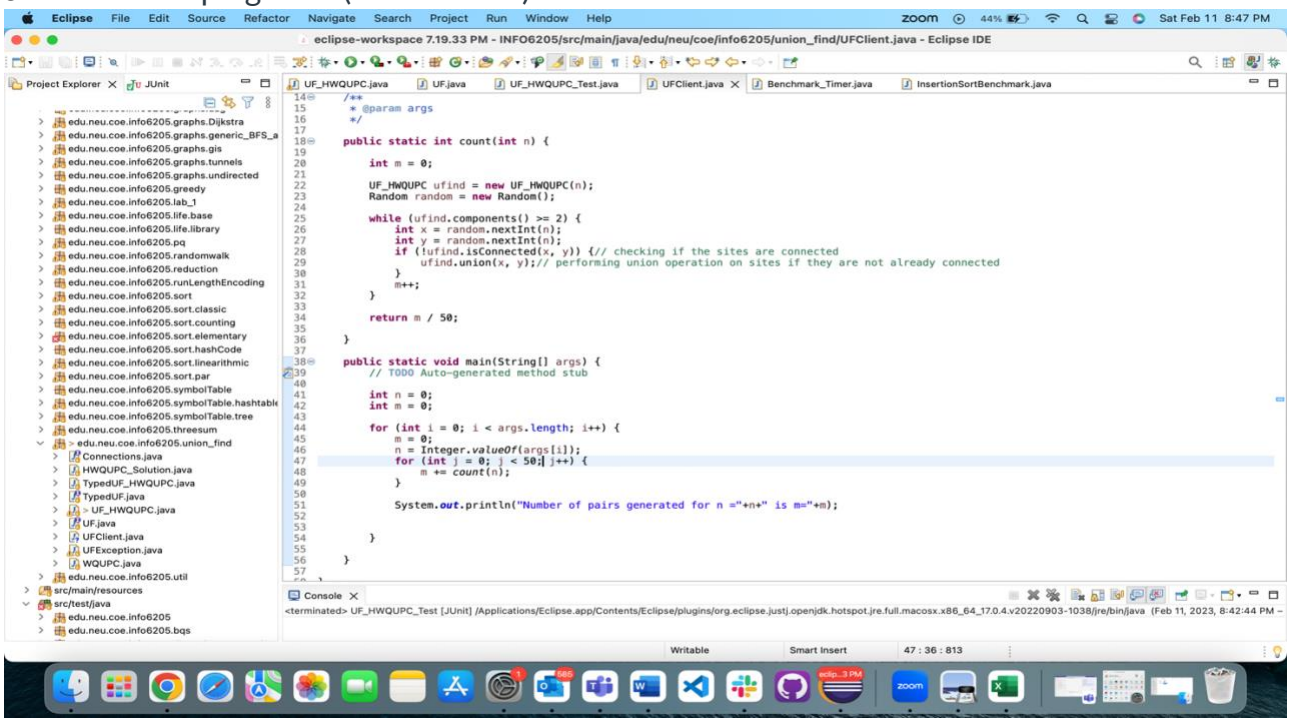
a) Implementing height-weighted Quick Union with Path Compression.



**b) Test Cases:**

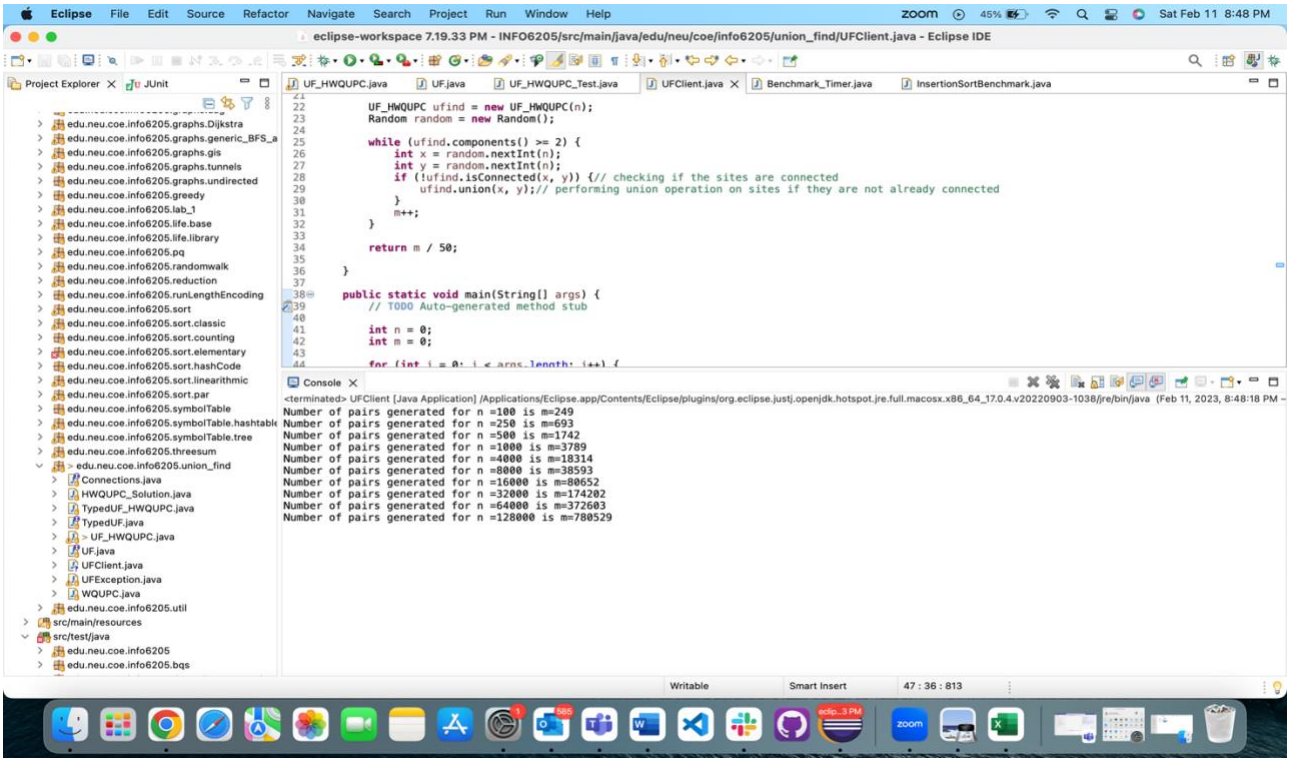



**STEP 2:** developing a UF ("union-find") client



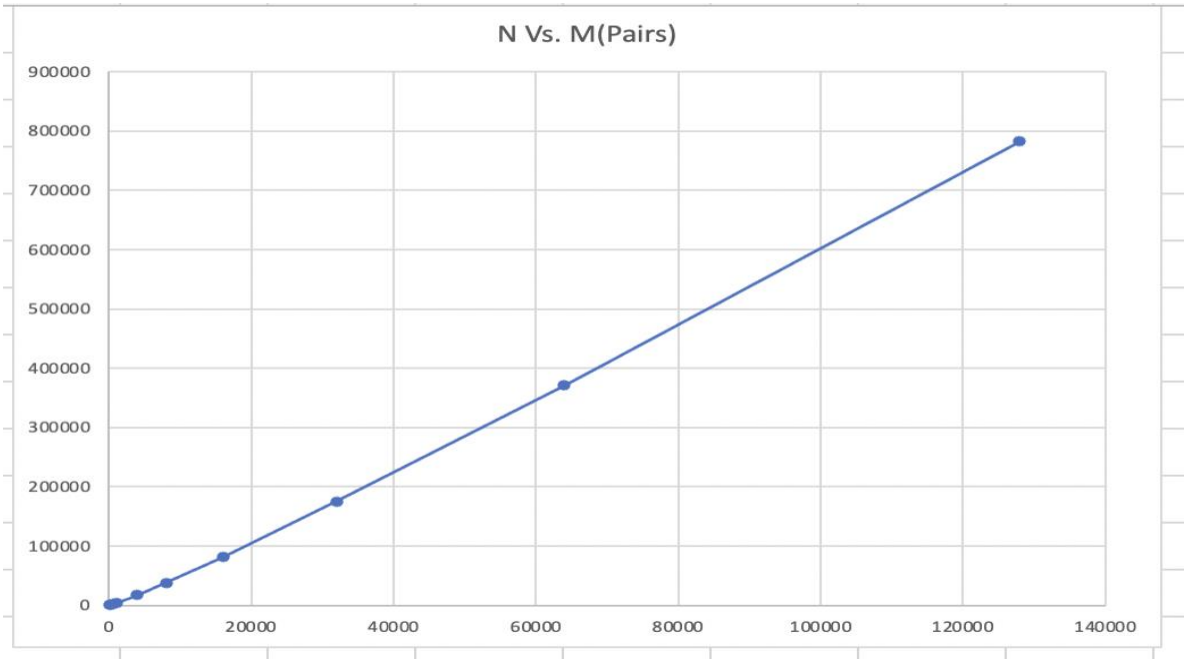
**Step 3:** Relationship between the number of objects (*n*) and the number of pairs (*m*)

**M=0.5*N*ln(N)**

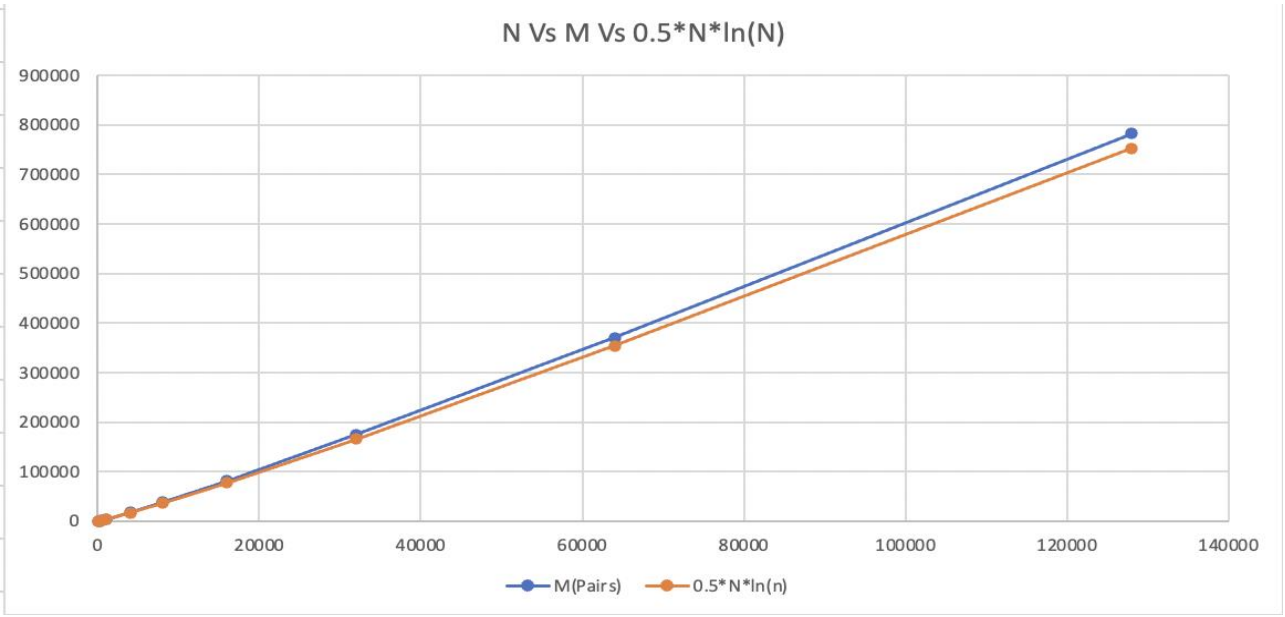



| | A | B | C | D |
|---|---|---|---|---|
| 1 | **N** | **M(Pairs)** | **N*ln(n)** | **0.5*N*ln(n)** |
| 2 | 100 | 240 | 460.5170186 | 230.2585093 |
| 3 | 250 | 750 | 1380.365229 | 690.1826147 |
| 4 | 500 | 1720 | 3107.304049 | 1553.652025 |
| 5 | 1000 | 3666 | 6907.755279 | 3453.877639 |
| 6 | 4000 | 17311 | 33176.19856 | 16588.09928 |
| 7 | 8000 | 37997 | 71897.57457 | 35948.78728 |
| 8 | 16000 | 81646 | 154885.504 | 77442.75201 |
| 9 | 32000 | 175295 | 331951.7178 | 165975.8589 |
| 10 | 64000 | 370732 | 708264.8552 | 354132.4276 |
| 11 | 128000 | 782193 | 1505252.549 | 752626.2747 |

**Graph: (N vs M)**




**N Vs M vs 0.5 *N*ln(N):**



**Relationship & Conclusion:**

From the Above two graphs and values we can clearly see that M is almost equal to 0.5 times of N*ln(N).

$$\mathbf{M=0.5*N*\ln(N)}$$