

Assignment 3

Name: Revanth Katha

NUID: 002981921

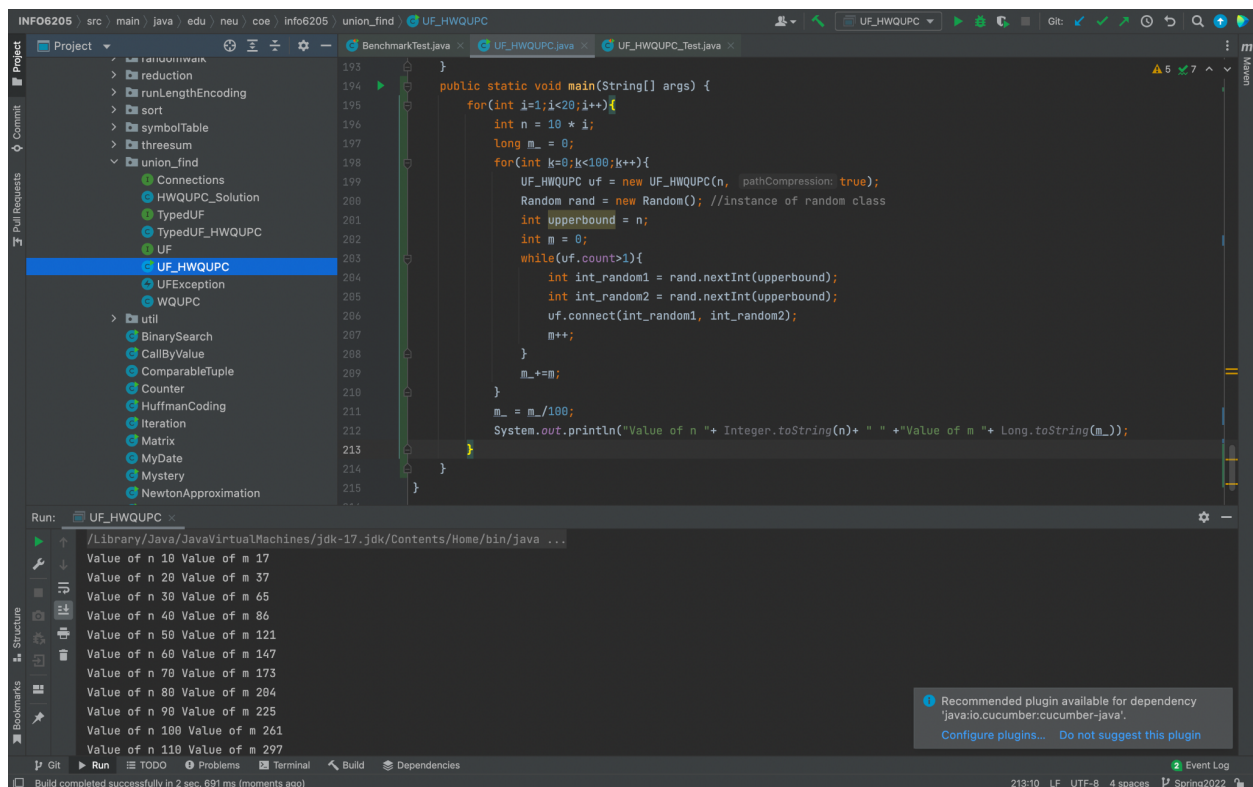
Q1.)

Task:

1)

- Implement height-weighted Quick Union with Path Compression.
- Check that the unit tests for this class all work.

Output:



The screenshot shows an IDE with the following components:

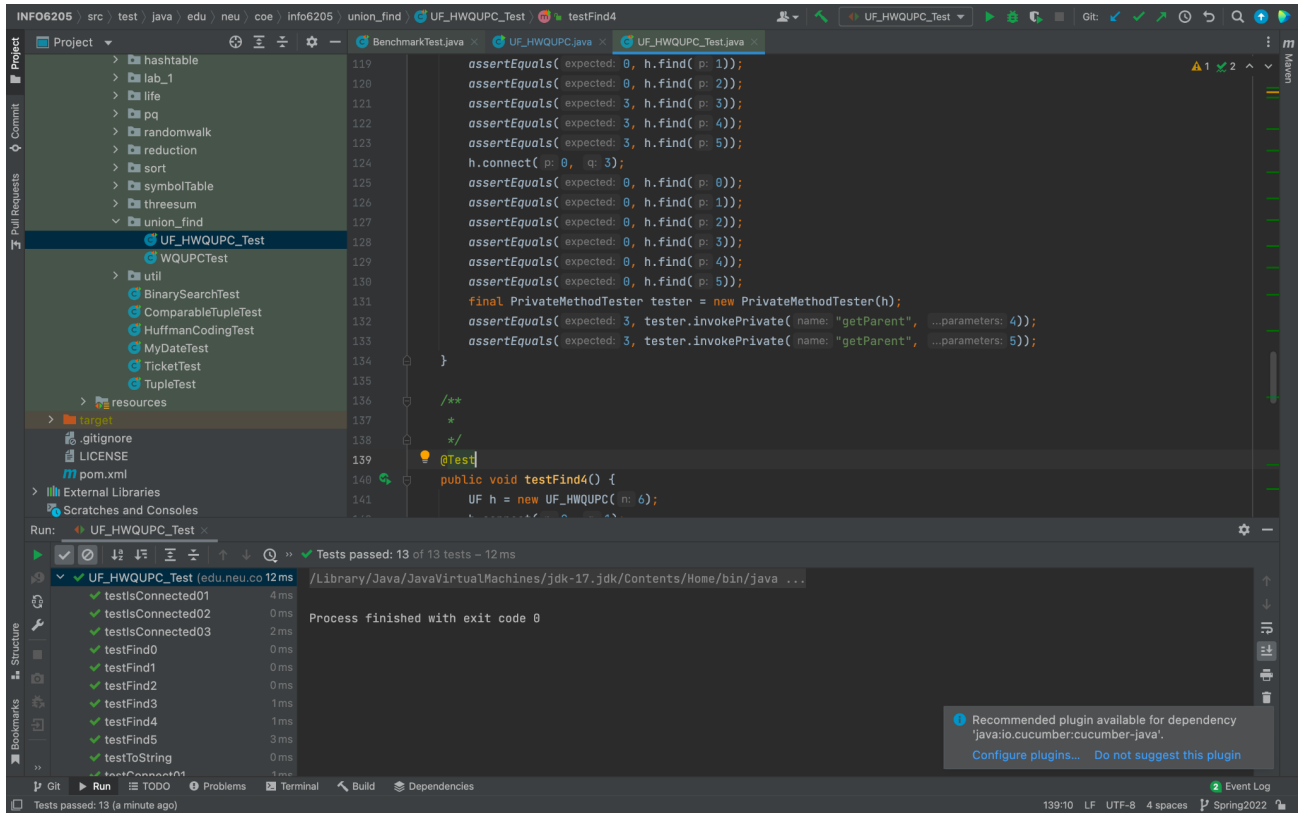
- Project Explorer:** Shows a project structure with a package named `union_find` containing classes `Connections`, `HWQUPC_Solution`, `TypedUF`, `TypedUF_HWQUPC`, `UF`, **`UF_HWQUPC`**, `UFException`, and `WQUPC`. There is also a `util` package with various utility classes.
- Code Editor:** Displays the `UF_HWQUPC` class with the following code:

```
193 }
194
195 public static void main(String[] args) {
196     for(int i=1;i<20;i++){
197         int n = 10 * i;
198         long m_ = 0;
199         for(int k=0;k<100;k++){
200             UF_HWQUPC uf = new UF_HWQUPC(n, pathCompression: true);
201             Random rand = new Random(); //instance of random class
202             int upperbound = n;
203             int m = 0;
204             while(uf.count>1){
205                 int int_random1 = rand.nextInt(upperbound);
206                 int int_random2 = rand.nextInt(upperbound);
207                 uf.connect(int_random1, int_random2);
208                 m++;
209             }
210             m_ += m;
211         }
212         m_ = m_/100;
213         System.out.println("Value of n "+ Integer.toString(n)+ " "+ "Value of m "+ Long.toString(m_));
214     }
215 }
```
- Run Console:** Shows the output of the program for `n` values from 10 to 110. The output is as follows:

```
Value of n 10 Value of m 17
Value of n 20 Value of m 37
Value of n 30 Value of m 65
Value of n 40 Value of m 86
Value of n 50 Value of m 121
Value of n 60 Value of m 147
Value of n 70 Value of m 173
Value of n 80 Value of m 204
Value of n 90 Value of m 225
Value of n 100 Value of m 261
Value of n 110 Value of m 297
```
- Bottom Bar:** Shows the status bar with "Build completed successfully in 2 sec, 691 ms (moments ago)", "213:10 LF UTF-8 4 spaces", and "Spring 2022".

Unit tests result: Following are the screenshots for running Quick Union with Path Compression test cases

Assignment 3

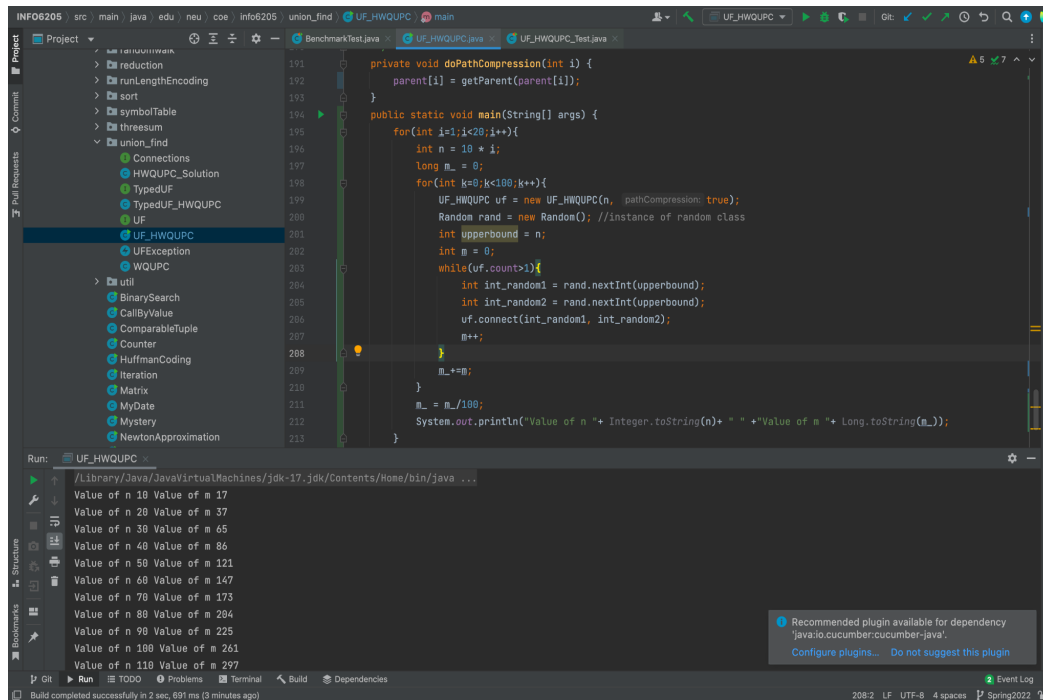


Q2.)

Task: Create a main program that takes n from the command line, calls count(), and prints the returned value.

Assignment 3

Output:



The screenshot shows an IDE with a project named 'union_find'. The code in 'UF_HWQUPC.java' implements a Union-Find algorithm with path compression. The 'main' method tests the algorithm for various values of n (from 10 to 110) and m (from 1 to 100). The output shows that m increases logarithmically with n .

```
private void doPathCompression(int i) {
    parent[i] = getParent(parent[i]);
}

public static void main(String[] args) {
    for(int i=1; i<20; i++){
        int n = 10 * i;
        long m_ = 0;
        for(int k=0; k<100; k++){
            UF_HWQUPC uf = new UF_HWQUPC(n, pathCompression: true);
            Random rand = new Random(); //instance of random class
            int upperbound = n;
            int m = 0;
            while(uf.count>1){
                int int_random1 = rand.nextInt(upperbound);
                int int_random2 = rand.nextInt(upperbound);
                uf.connect(int_random1, int_random2);
                m++;
            }
            m_ += m;
        }
        m_ = m_/100;
        System.out.println("Value of n " + Integer.toString(n) + " " + "Value of m " + Long.toString(m_));
    }
}
```

Run: UF_HWQUPC

```
/Library/Java/JavaVirtualMachines/jdk-17.jdk/Contents/Home/bin/java ...
Value of n 10 Value of m 17
Value of n 20 Value of m 37
Value of n 30 Value of m 65
Value of n 40 Value of m 86
Value of n 50 Value of m 121
Value of n 60 Value of m 147
Value of n 70 Value of m 173
Value of n 80 Value of m 204
Value of n 90 Value of m 225
Value of n 100 Value of m 261
Value of n 110 Value of m 297
```

Q3.)

Task: Determine the relationship between the number of objects (n) and the number of pairs (m)

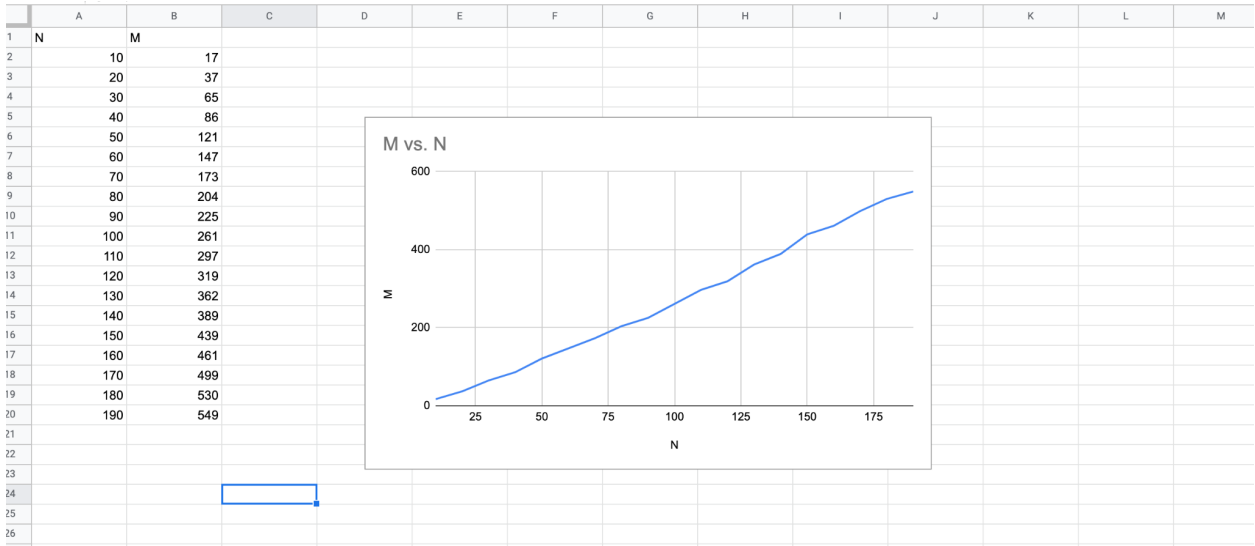
Conclusion:

As the number of objects(n) increases the number of pairs(m) increases n logarithmic times

$$m = n * \log n$$

Assignment 3

Evidence: Following are the evidence to prove the relationship



Assignment 3

N	M
10	17
20	37
30	65
40	86
50	121
60	147
70	173
80	204
90	225
100	261
110	297
120	319
130	362
140	389
150	439
160	461
170	499
180	530
190	549