

Program Structures and Algorithms

Fall 2022(SEC 06)

NAME: Olasunkanmi Olayinka

NUID: 001512266

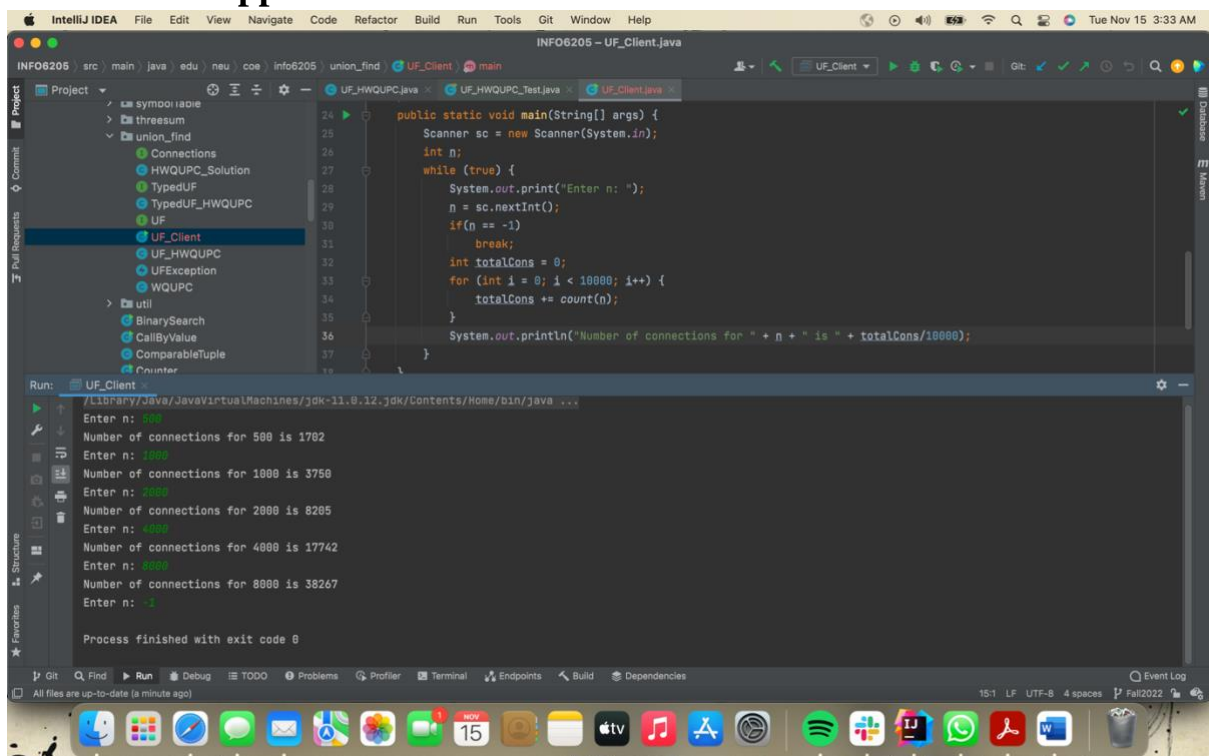
Task:

- Implement height-weighted Quick Union with Path Compression
- Use my implementation of UF_HWQUPC, develop a UF client that takes an integer value of n from the command line to determine the number of “sites”
- Determine the relationship between the number of objects (n) and the number of pairs (m) generated to accomplish this.

Relationship Conclusion:

The number of pairs(m) is approximately $n + n \log(n)$, where n is the number of object/nodes

Evidence to support that conclusion:



The screenshot shows the IntelliJ IDEA IDE with the project 'INFO6205' open. The file 'UF_Client.java' is selected in the Project view. The code in the editor is as follows:

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int n;
    while (true) {
        System.out.print("Enter n: ");
        n = sc.nextInt();
        if (n == -1)
            break;
        int totalCons = 0;
        for (int i = 0; i < 10000; i++) {
            totalCons += count(n);
        }
        System.out.println("Number of connections for " + n + " is " + totalCons/10000);
    }
}
```

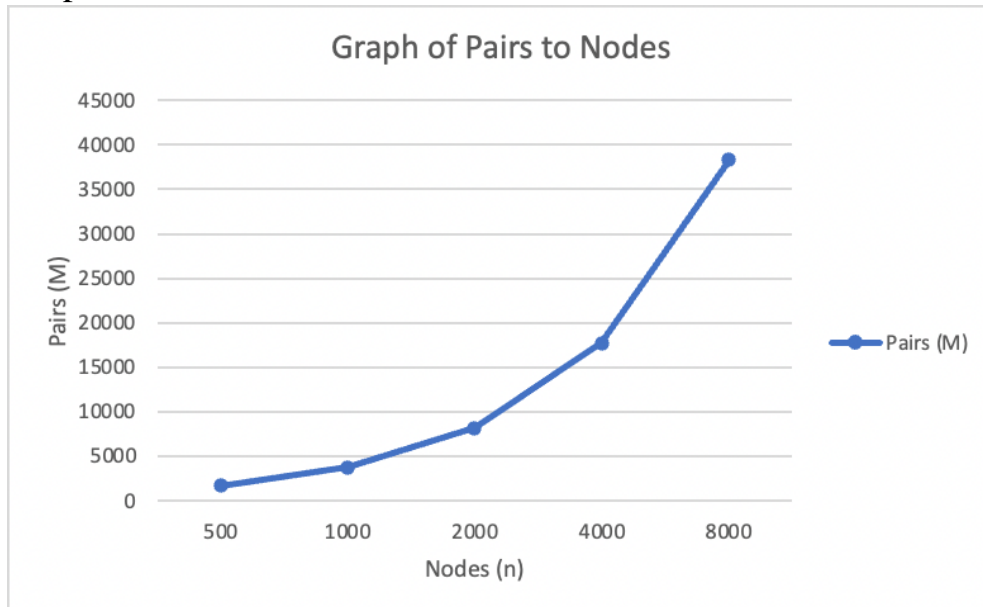
The Run window shows the output of the program:

```
Enter n: 500
Number of connections for 500 is 1782
Enter n: 1000
Number of connections for 1000 is 3758
Enter n: 2000
Number of connections for 2000 is 8285
Enter n: 4000
Number of connections for 4000 is 17742
Enter n: 8000
Number of connections for 8000 is 38267
Enter n: -1
Process finished with exit code 0
```

Nodes (n)	Pairs (M)	$n \cdot \lg(n)$
500	1702	1349.485
1000	3750	3000
2000	8205	6602.05999
4000	17742	14408.24
8000	38267	31224.7199

Graphical Representation:

Graph of Pairs to Nodes



Unit Test Screenshots:

