Program Structures and Algorithms
Spring 2023(SEC – 1)

NAME: Foram Kamani

NUID: 002732551

Task:

Step 1: (a) Implement height-weighted Quick Union with Path Compression. For this, you will flesh out the class UF_HWQUPC. All you have to do is to fill in the sections marked with // TO BE IMPLEMENTED ... // ...END IMPLEMENTATION.

(b) Check that the unit tests for this class all work. You must show "green" test results in your submission (the screenshot is OK).

Step 2:

Using your implementation of UF_HWQUPC, develop a UF ("union-find") client that takes an integer value n from the command line to determine the number of "sites." Then generates random pairs of integers between 0 and n-1, calling connected() to determine if they are connected and union() if not. Loop until all sites are connected then print the number of connections generated. Package your program as a static method count() that takes n as the argument and returns the number of connections; and a main() that takes n from the command line, calls count(), and prints the returned value. If you prefer, you can create a main program that doesn't require any input and runs the experiment for a fixed set of n values. Show evidence of your run(s).

Step 3:

Determine the relationship between the number of objects (*n*) and the number of pairs (*m*) generated to accomplish this (i.e. to reduce the number of components from *n* to 1). Justify your conclusion in terms of your observations and what you think might be going on.

Relationship Conclusion:

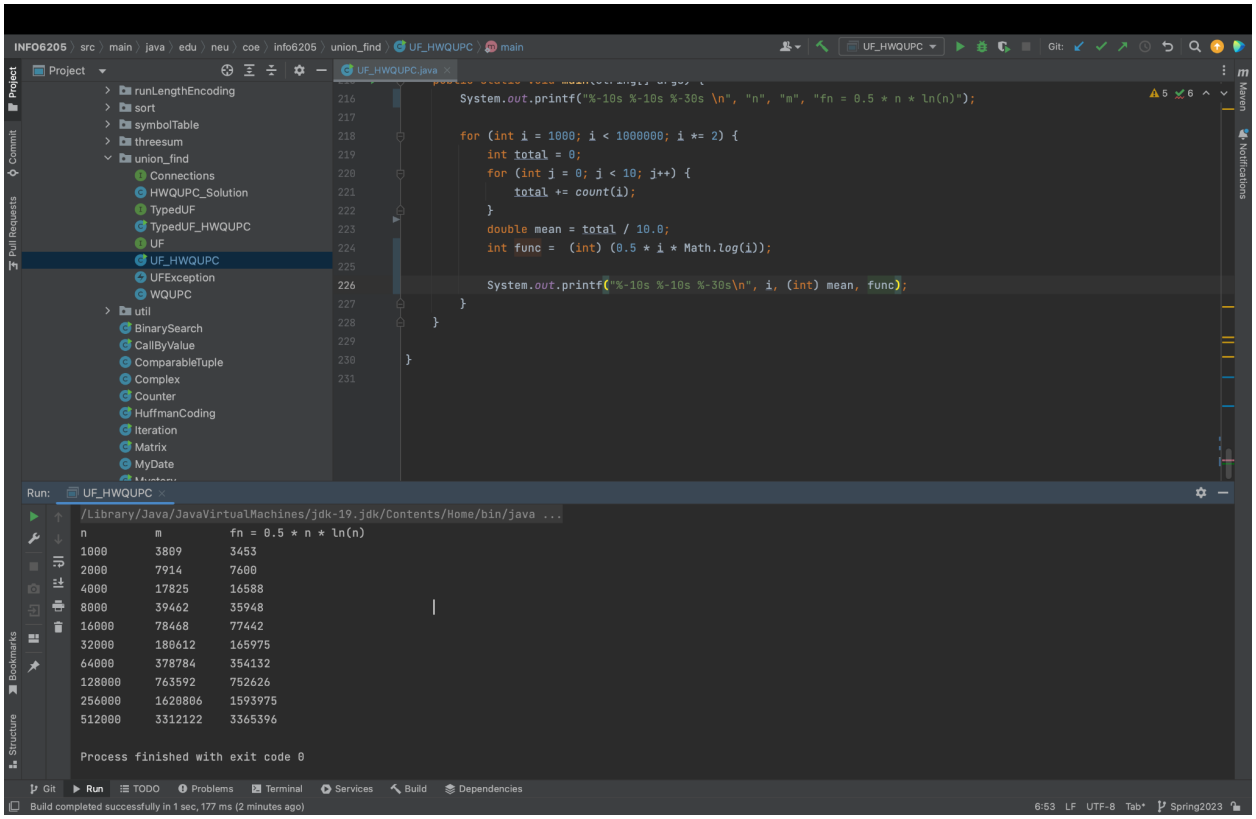
After executing the main method with multiple values of n, I have concluded that the relationship between the number of objects(n) and the number of pairs(m) generated is nearly equal to m=n(ln(n))/2

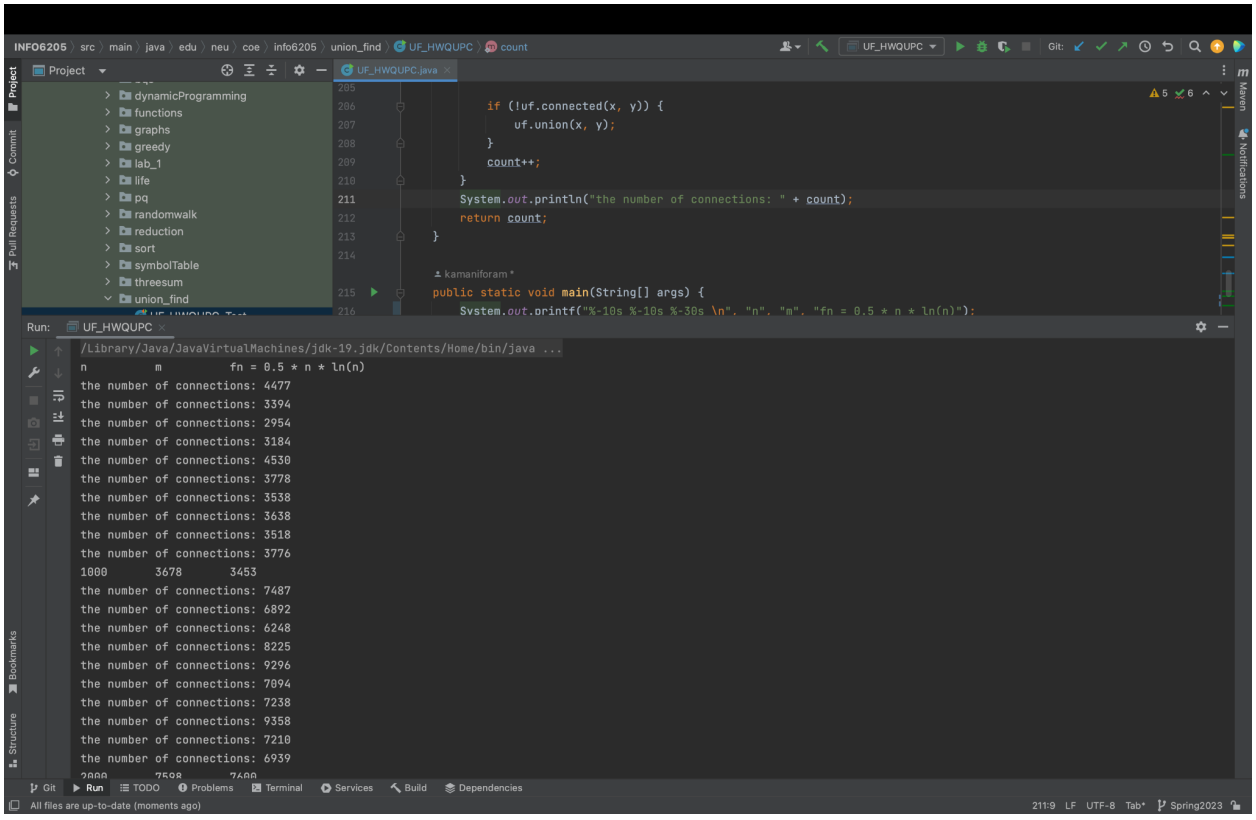
$$\mathbf{m = f(n) = 0.5 * n * ln(n)}$$

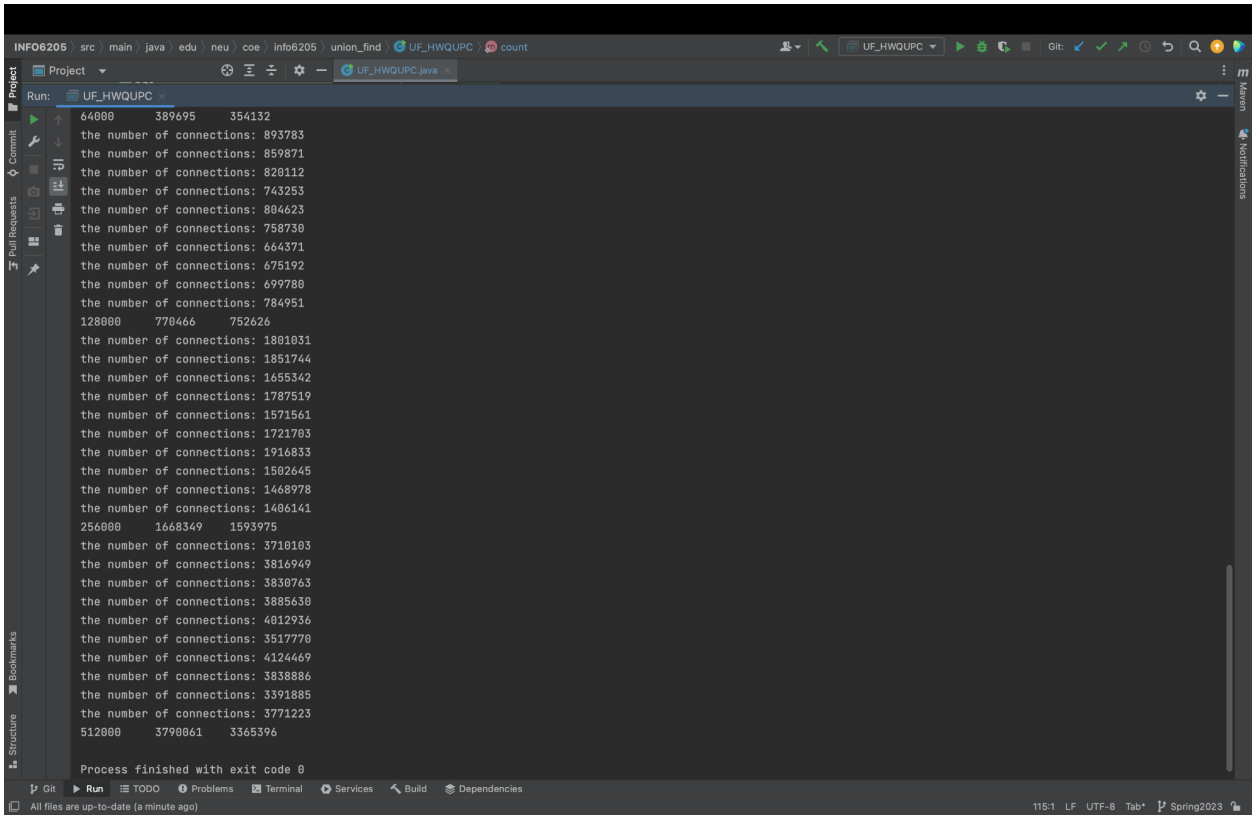
Evidence to support that conclusion:

1. Values of n are ranging from 1000 to 512000(doubling each time).

2. For each value of n, the program is running 100 times, and the average value of m is shown below.

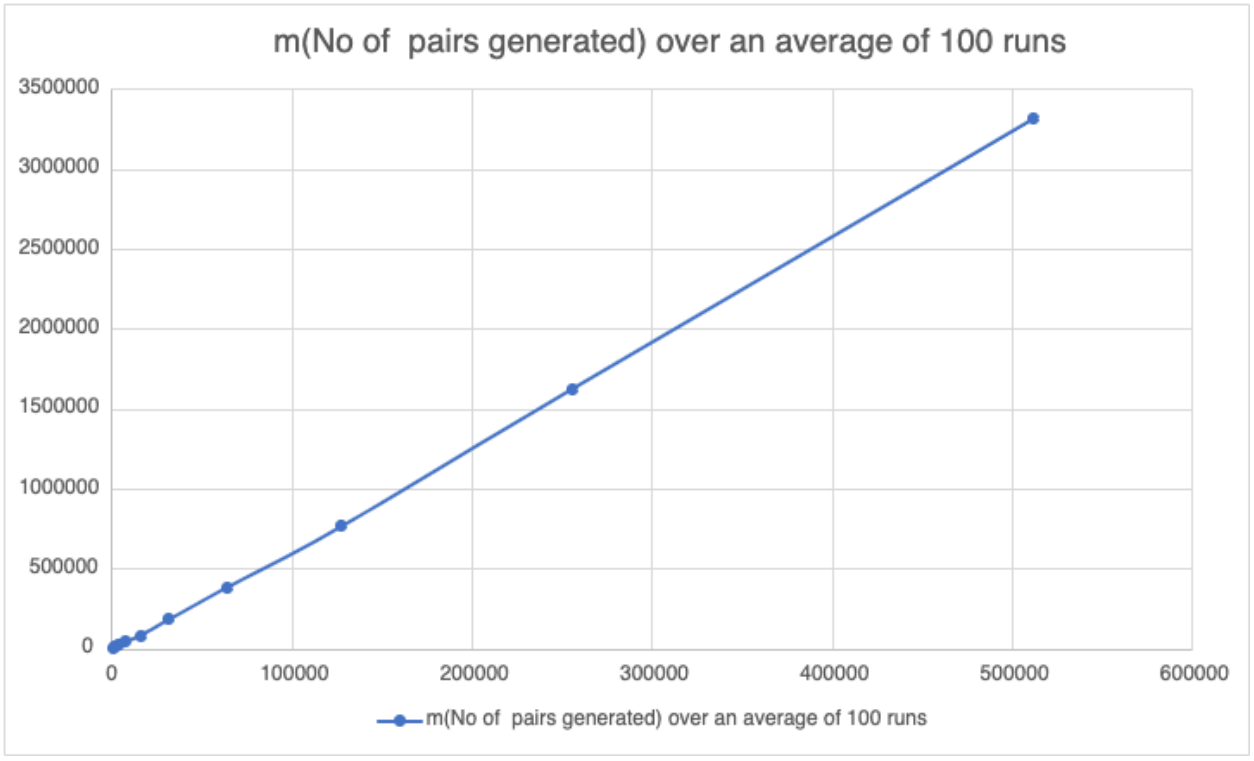



INFO6205 src main java edu neu coe info6205 union_find UF_HWQUPC count
UF_HWQUPC
Git:
Project
UF_HWQUPC.java
dynamicProgramming
functions
graphs
greedy
lab_1
life
pq
randomwalk
reduction
sort
symbolTable
threesum
union_find
if (!uf.connected(x, y)) {
uf.union(x, y);
}
count++;
}
System.out.println("the number of connections: " + count);
return count;
}
kamaniforam *
public static void main(String[] args) {
System.out.printf("%-10s %-10s %-30s \n", "n", "m", "fn = 0.5 * n * ln(n)");
Run:
UF_HWQUPC
/Library/Java/JavaVirtualMachines/jdk-19.jdk/Contents/Home/bin/java ...
n m fn = 0.5 * n * ln(n)
the number of connections: 4477
the number of connections: 3394
the number of connections: 2954
the number of connections: 3184
the number of connections: 4530
the number of connections: 3778
the number of connections: 3538
the number of connections: 3638
the number of connections: 3518
the number of connections: 3776
1000 3678 3453
the number of connections: 7487
the number of connections: 6892
the number of connections: 6248
the number of connections: 8225
the number of connections: 9296
the number of connections: 7094
the number of connections: 7238
the number of connections: 9358
the number of connections: 7210
the number of connections: 6939
Git Run TODO Problems Terminal Services Build Dependencies
All files are up-to-date (moments ago)
211:9 LF UTF-8 Tab* Spring2023


INFO6205 src main java edu neu coe info6205 union_find UF_HWQUPC count
UF_HWQUPC
Git:
Project
UF_HWQUPC.java
Run:
UF_HWQUPC
64000 389695 354132
the number of connections: 893783
the number of connections: 859871
the number of connections: 820112
the number of connections: 743253
the number of connections: 804623
the number of connections: 758730
the number of connections: 664371
the number of connections: 675192
the number of connections: 699780
the number of connections: 784951
128000 770466 752626
the number of connections: 1801031
the number of connections: 1851744
the number of connections: 1655342
the number of connections: 1787519
the number of connections: 1571561
the number of connections: 1721703
the number of connections: 1916833
the number of connections: 1502645
the number of connections: 1468978
the number of connections: 1406141
256000 1668349 1593975
the number of connections: 3710103
the number of connections: 3816949
the number of connections: 3830763
the number of connections: 3885630
the number of connections: 4012936
the number of connections: 3517770
the number of connections: 4124469
the number of connections: 3838886
the number of connections: 3391885
the number of connections: 3771223
512000 3790061 3365396
Process finished with exit code 0
Git Run TODO Problems Terminal Services Build Dependencies
All files are up-to-date (a minute ago)
115:1 LF UTF-8 Tab* Spring2023

Graphical Representation:

n(No of Objects)	m(No of pairs generated) over an average of 100 runs	fn = 0.5 * n * ln(n)
1000	3809	3453
2000	7914	7600
4000	17825	16588
8000	39462	35948
16000	78468	77442
32000	180612	165975
64000	378784	354132
128000	763592	752626
256000	1620806	1593975
512000	3312122	3365396



Unit Test Screenshots: