Varun Venkatesh Gowda (002126161)

Program Structures & Algorithms

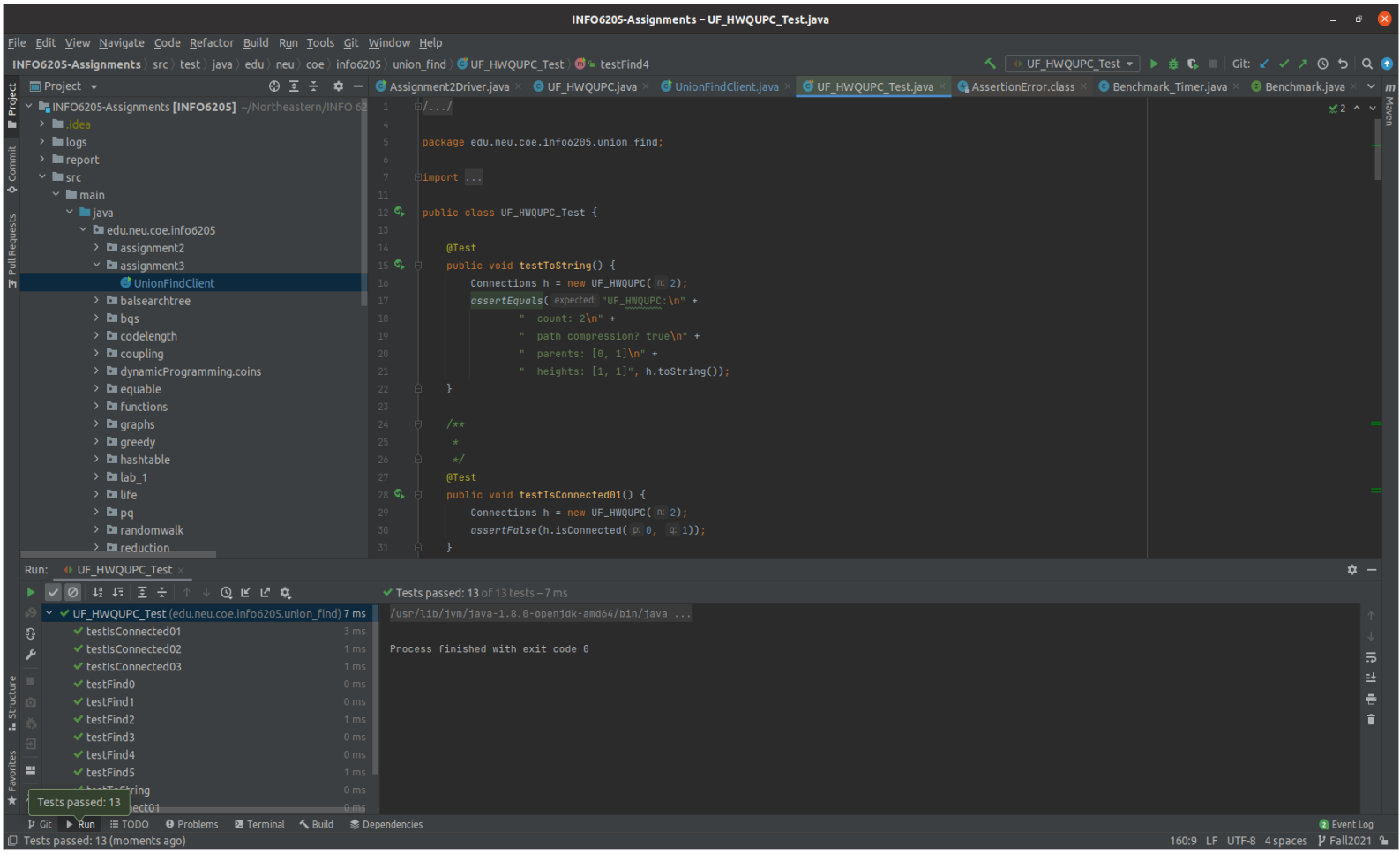
Fall 2021

Assignment No. 3

Tasks:

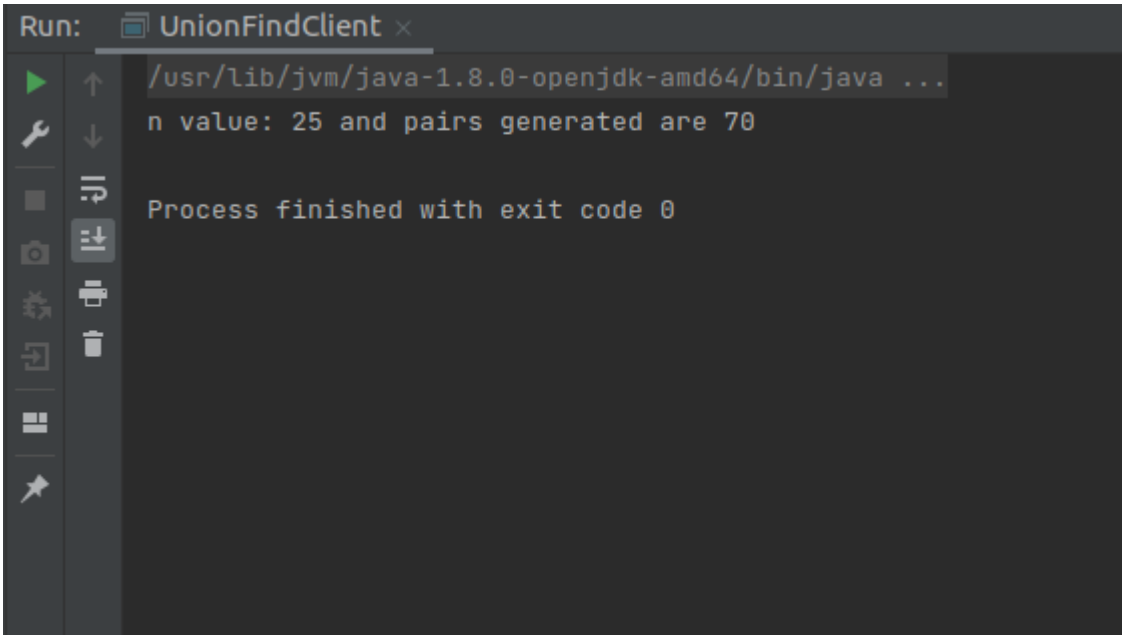
1. Implemented find(int p) method in UF_HWQUPC.java
2. Implemented mergeComponents(int i, int j) in UF_HWQUPC.java
3. Implemented doPathCompression(int i) in UF_HWQUPC.java
4. Ran test cases from UF_HWQUPC_Test.java (all passed)
5. Coded UnionFindClient.java in **edu.neu.coe.info6205.assignment3** package to run the experiments
6. Implemented a main() method to run the experiments for defined values of n
7. Implemented count() method calculate and return the pairs generated to reduce the components to 1
8. Ran the experiments for multiple values of “n” and 3 runs for each value of “n”
9. Tabulated the readings in excel and generated a graph to observe the trend

Tests passing screenshot:

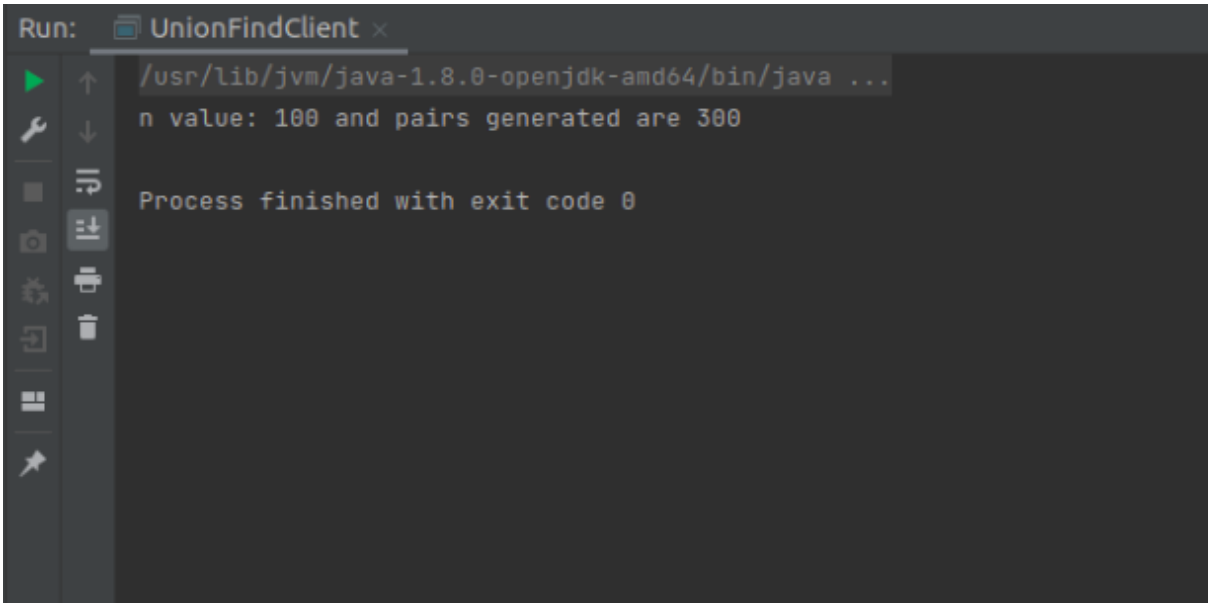


Outputs:

For n = 25

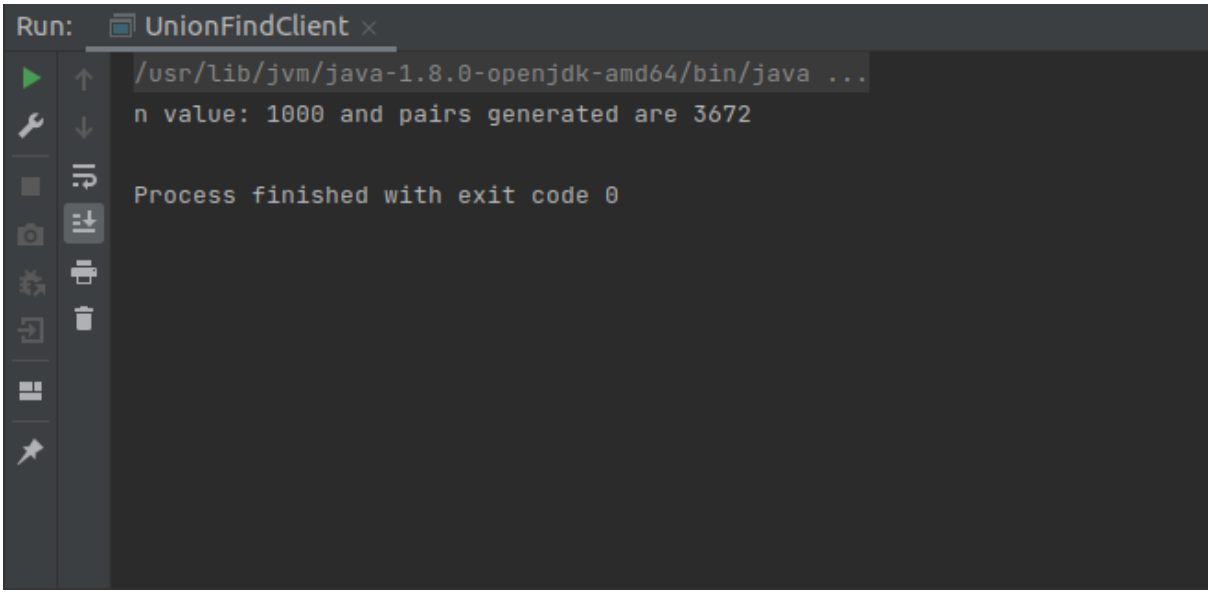


For n = 100

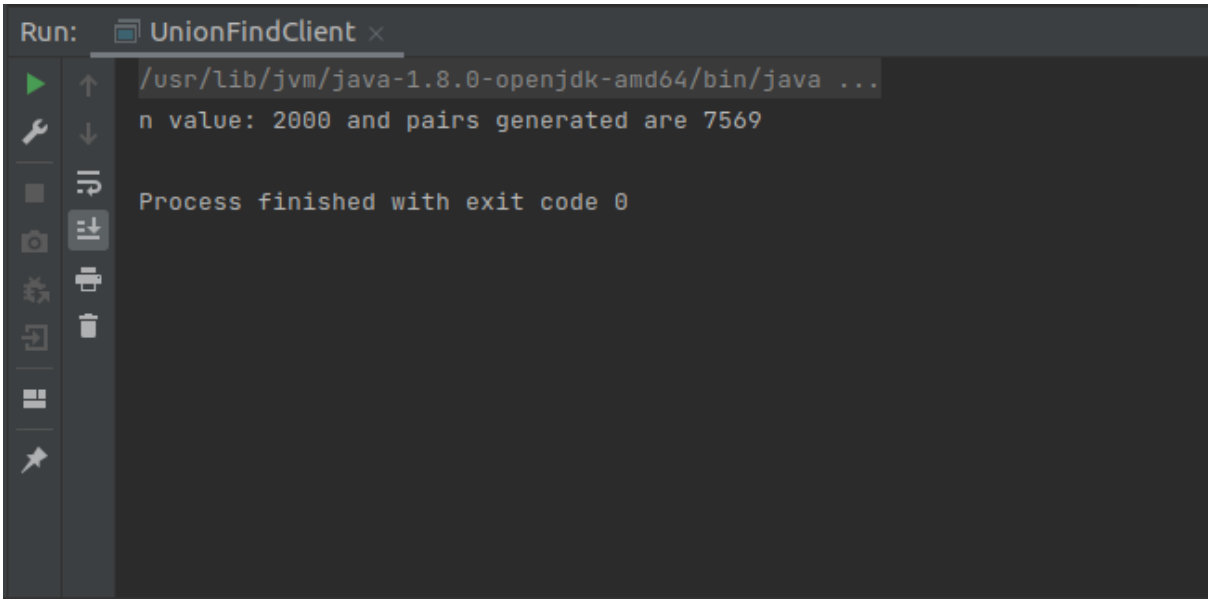


For n = 500

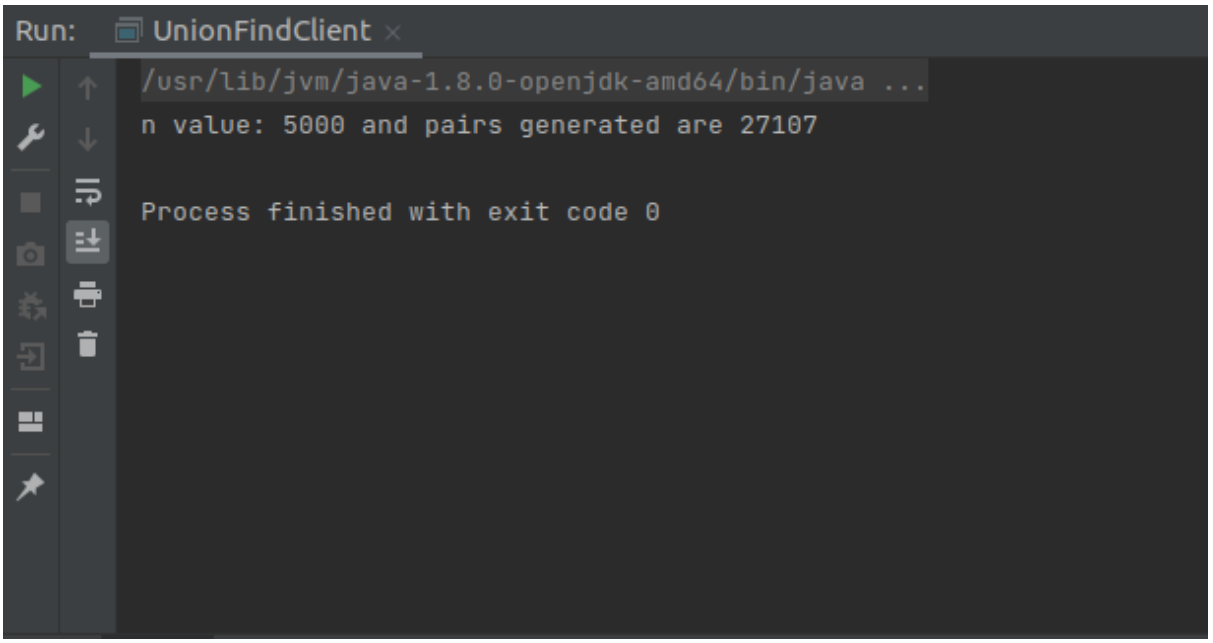
For n = 1000



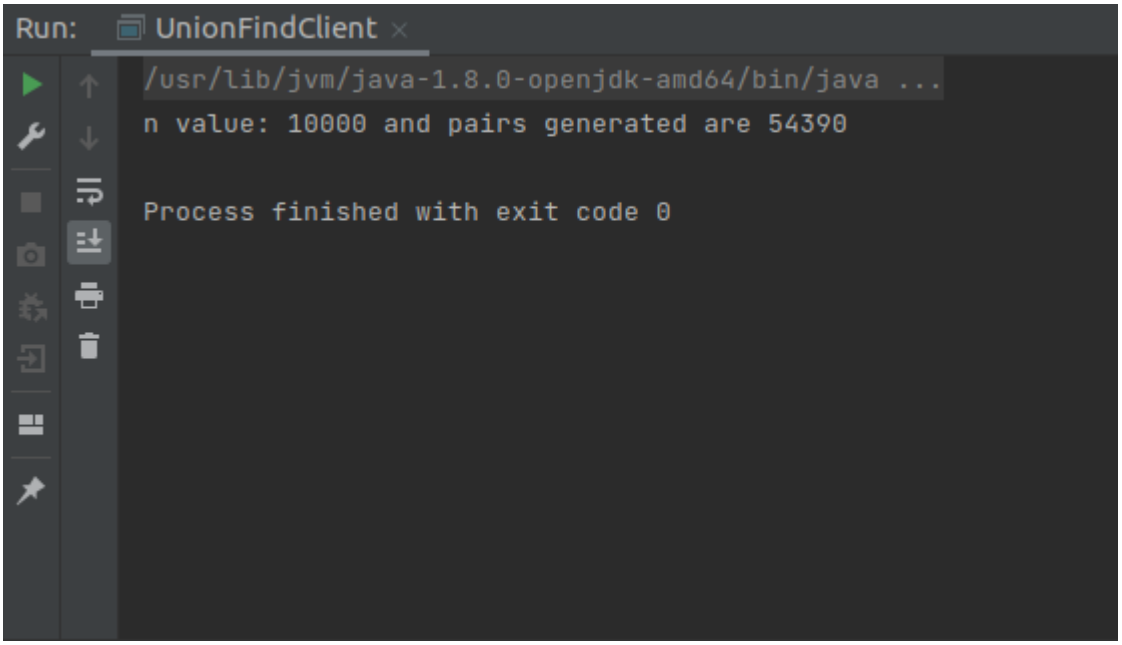
For n = 2000



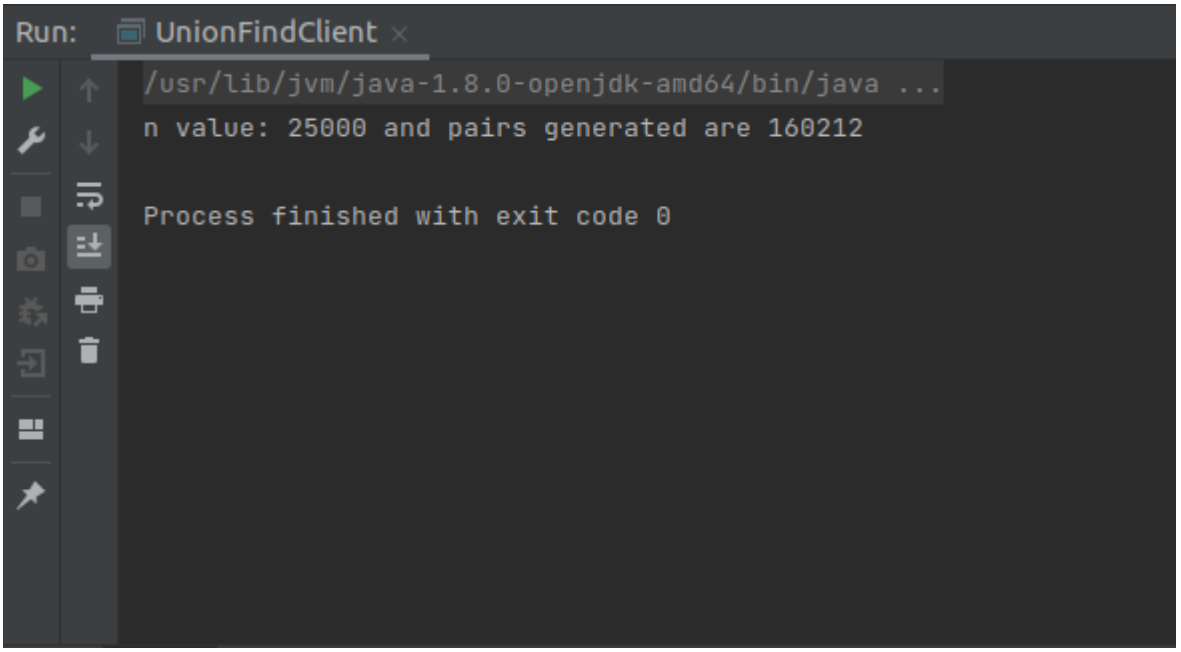
For n = 5000



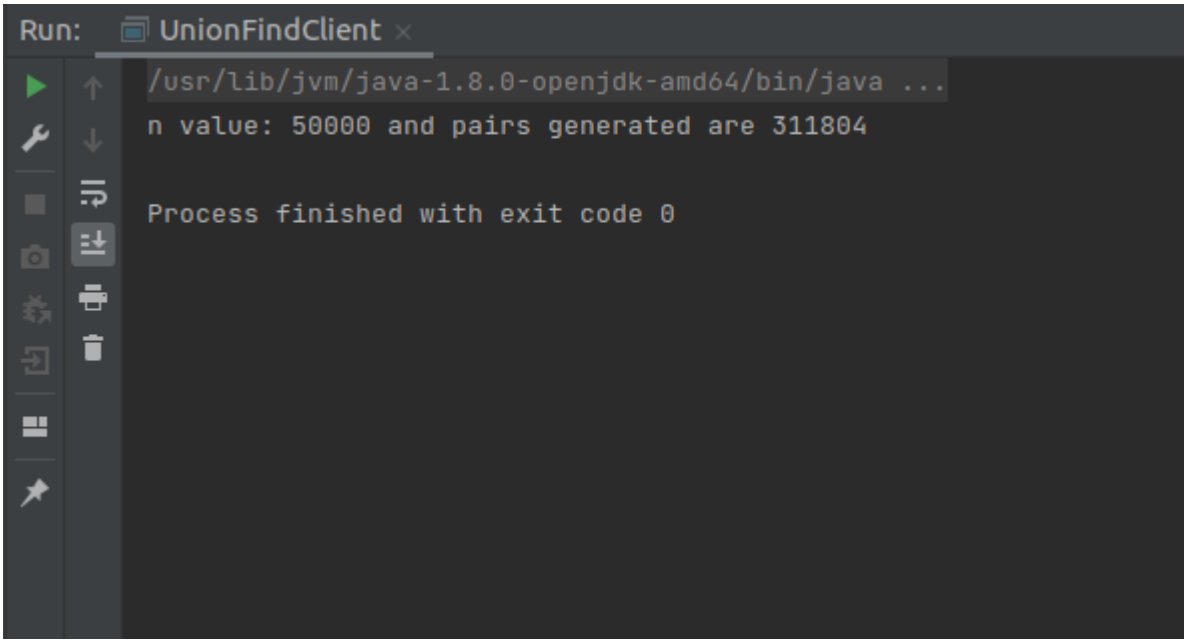
For n = 10000



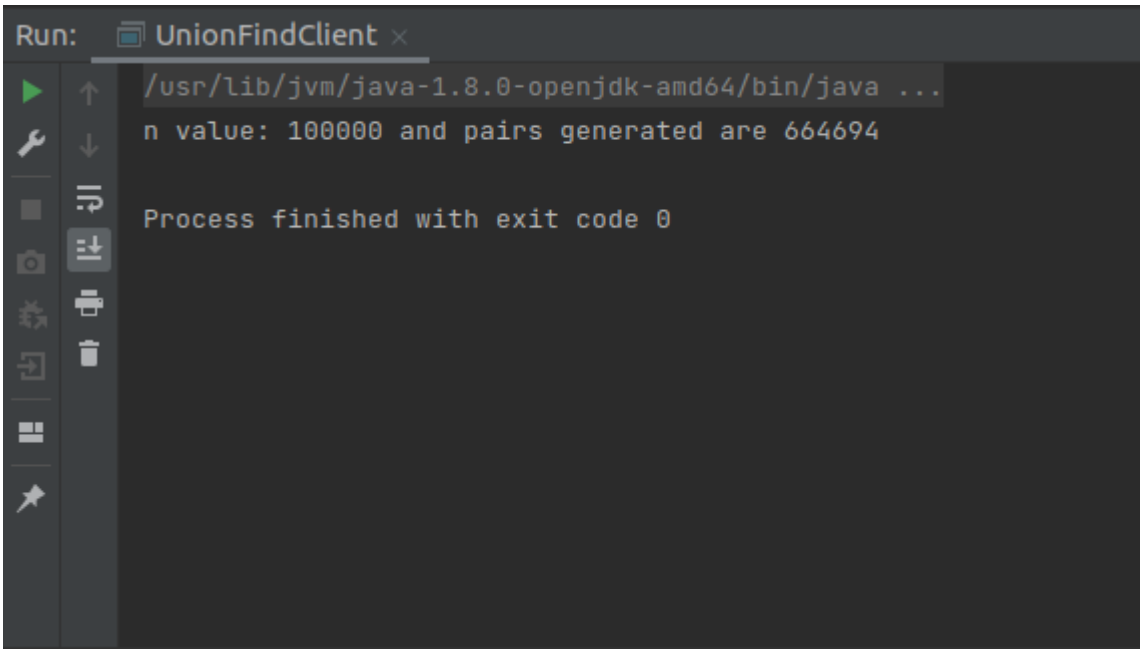
For n = 25000



For n = 50000

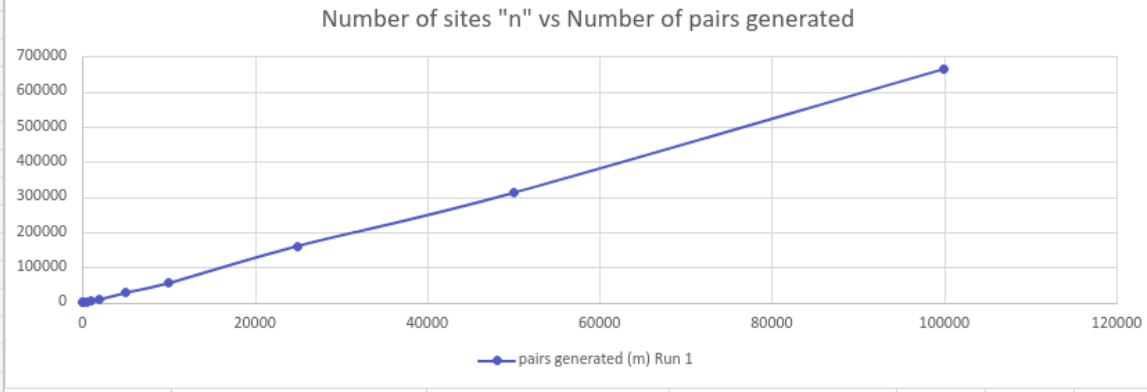


For n = 100000



Tabulation/ Graph:

	A	B	C	D	E	F	G
1	number of sites (n)	pairs generated (m) Run 1	pairs generated (m) Run 2	pairs generated (m) Run 3			
2	25	70	103	51			
3	100	300	233	311			
4	500	1658	1616	1580			
5	1000	3672	2764	4641			
6	2000	7569	8043	8033			
7	5000	27107	22879	25462			
8	10000	54390	51030	39773			
9	25000	160212	138885	122170			
10	50000	311804	244769	261728			
11	100000	664694	562899	702448			
12							



Conclusion:

From the readings obtained, it is seen that as the number of sites (n) increases, the number of pairs generated (m) also increases (Exponentially large in the case of larger values of n) and generates a linear graph. so I believe that “n” and “m” are directly proportional.

$$n \propto m$$