YanXuan NUID:001563047

**Program Structures & Algorithms**

**Fall 2021**

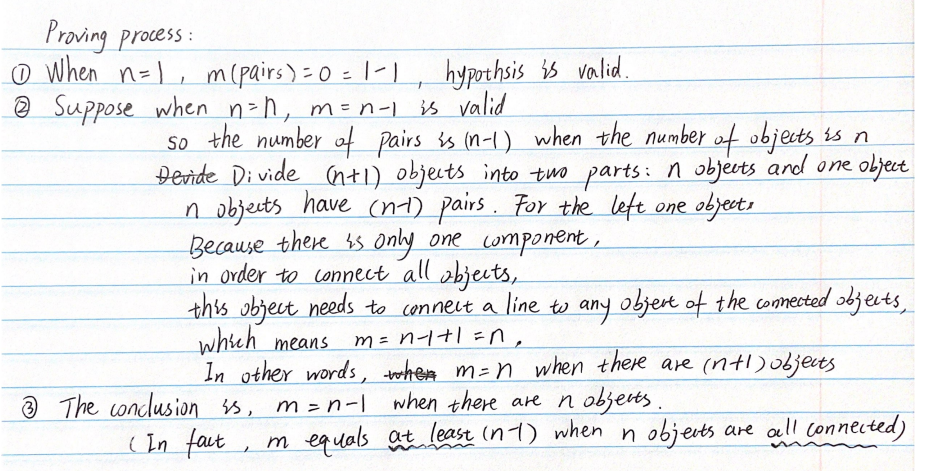
**Assignment No. 3**

* **Task (List down the tasks performed in the Assignment)**

1. **Implemented height-weighted Quick Union with Path Compression.**
2. **Used the implementation of UF\_HWQUPC, developed a UF ("union-find") client.**
3. **Perform data simulation, change the number of n("sites") and make experimental data tables and graphs.**
4. **Analyze and summarize the experimental data, and get the relationship between the number of objects (n) and the number of pairs (m).**

* **Relationship Conclusion: (For ex : z = a \* b)**

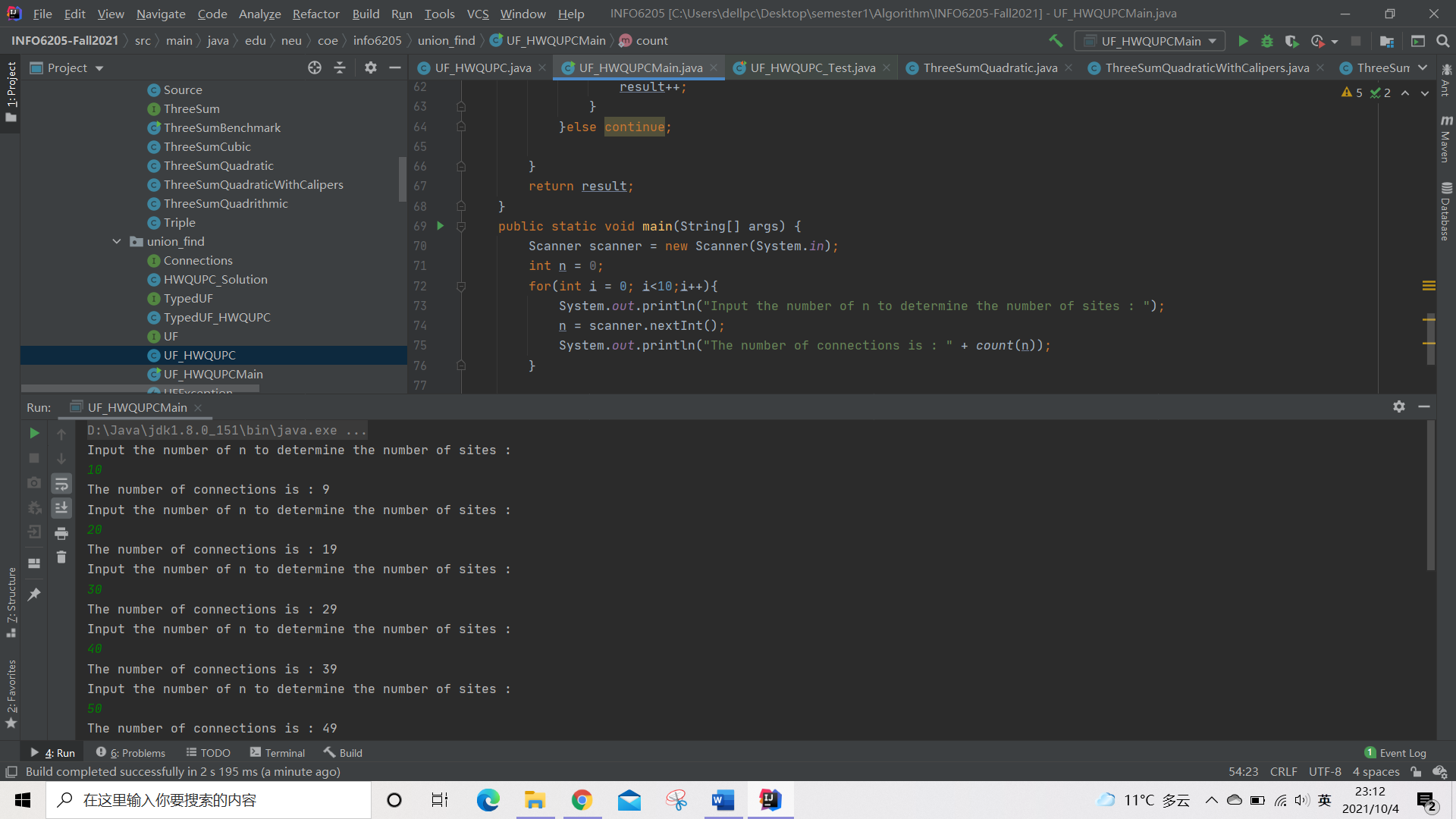
**The derivation process：**

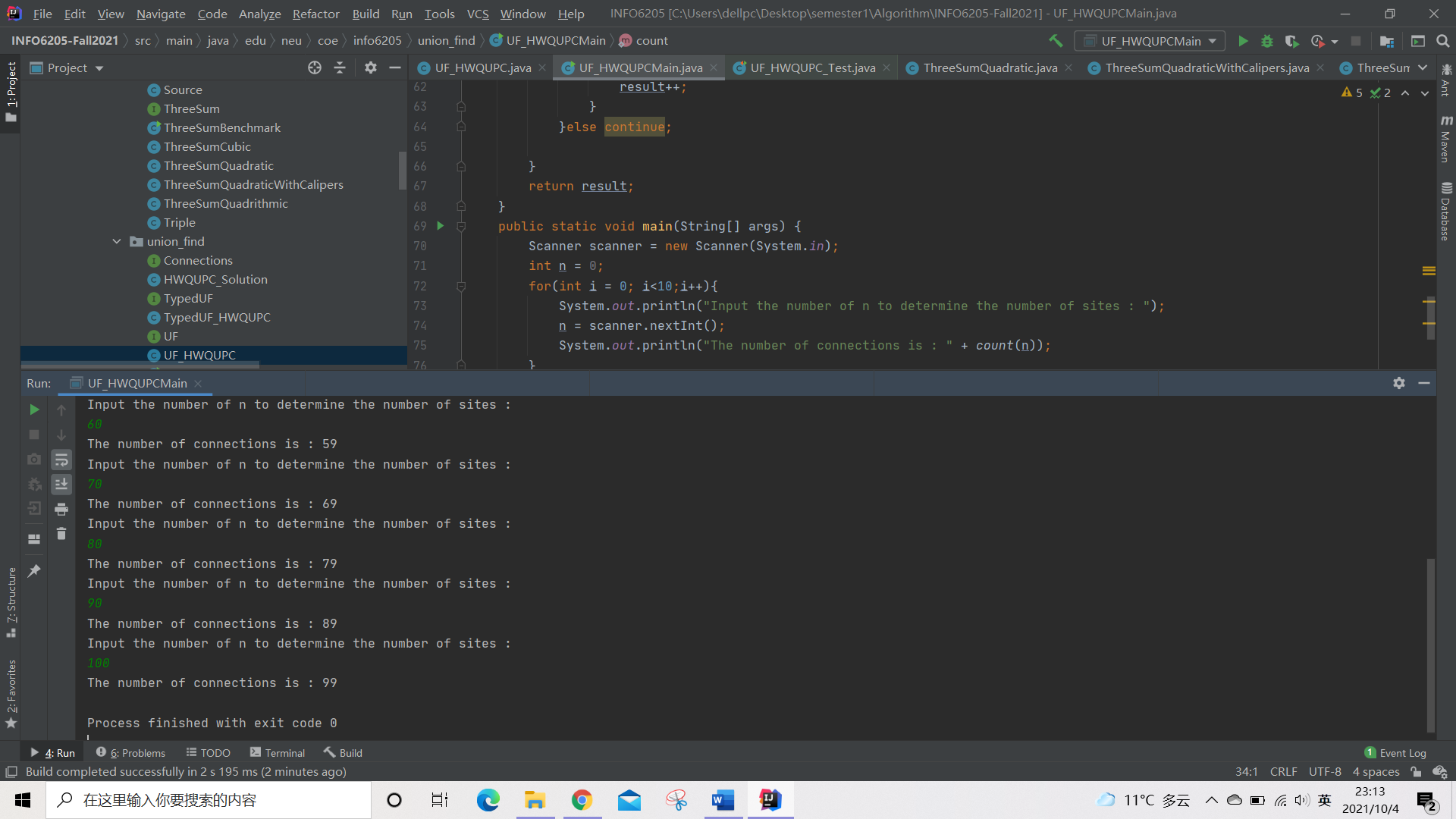


* **Evidence to support the conclusion:**

**\*\* The UF\_HWQUPCMain.java is in \src\main\java\edu\neu\coe\info6205\union\_find**

1. **Output (Snapshot of Code output in the terminal)**





**Output text:**

|  |
| --- |
| Input the number of n to determine the number of sites :  10  The number of connections is : 9  Input the number of n to determine the number of sites :  20  The number of connections is : 19  Input the number of n to determine the number of sites :  30  The number of connections is : 29  Input the number of n to determine the number of sites :  40  The number of connections is : 39  Input the number of n to determine the number of sites :  50  The number of connections is : 49  Input the number of n to determine the number of sites :  60  The number of connections is : 59  Input the number of n to determine the number of sites :  70  The number of connections is : 69  Input the number of n to determine the number of sites :  80  The number of connections is : 79  Input the number of n to determine the number of sites :  90  The number of connections is : 89  Input the number of n to determine the number of sites :  100  The number of connections is : 99  Process finished with exit code 0 |

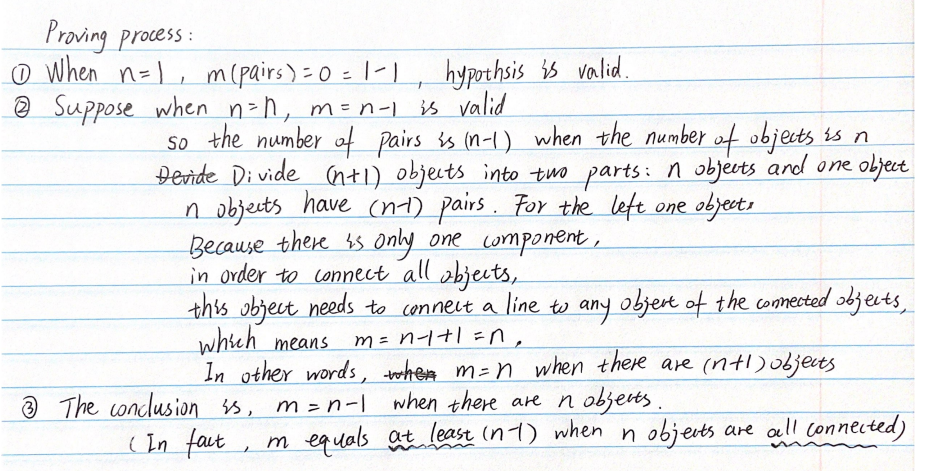
**Code**

public static int count(int n){  
 UF\_HWQUPC h = new UF\_HWQUPC(n);  
 Random r = new Random();  
 int result = 0;  
 int temp1 = 0;  
 int temp2 = 0;  
 while(true){  
 if(h.components() == 1) break;  
 temp1 = r.nextInt(n);  
 temp2 = r.nextInt(n);  
 if(temp1 != temp2){  
 if(!h.connected(temp1, temp2)){  
 h.connect(temp1, temp2);  
 result++;  
 }  
 }else continue;  
  
 }  
 return result;  
}  
public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 int n = 0;  
 for(int i = 0; i<10;i++){  
 System.*out*.println("Input the number of n to determine the number of sites : ");  
 n = scanner.nextInt();  
 System.*out*.println("The number of connections is : " + *count*(n));  
 }  
}

1. **Graphical Representation(Observations from experiments should be tabulated and analyzed by plotting graphs(usually in excel) to arrive on the relationship conclusion)**



**The derivation process：**



* **Unit tests result:(Snapshot of successful unit test run)**

