**Zhilong Zhu: 001525601**

**Program Structures & Algorithms**

**Spring 2021**

**Assignment 3**

* **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 (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.
* Don't forget to follow the submission guidelines. And to use sufficient (and sufficiently large) different values of n.
* **Output**

**文本

描述已自动生成**

* **Relationship Conclusion:**

m =

* **Evidence to support the conclusion:**

n1 = 10, m1 = 16, E(m1) = 12

n2 = 50, m2 = 114, E(m2) = 98

n3 = 250, m3 = 768, E(m3) = 690

n4 = 1250, m4 = 4780, E(m4) = 4457

n5 = 6250, m5 = 29012, E(m5) = 27314

n6 = 31250, m6 = 171384, E(m6) = 161715

* **Graphical representation:**

图表, 折线图

描述已自动生成

* **Unit tests result:**

图形用户界面, 文本, 应用程序

描述已自动生成