Haoyang Hu (001005743)

**Program Structures & Algorithms**

**Fall 2021**

**Assignment No. 03**

* **Task**

Step 1:  
(a) Implement height-weighted Quick Union with Path Compression.

(b) Check that the unit tests for this class all work.

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:**
* **Evidence to support the conclusion:**

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

**电脑萤幕的截图

描述已自动生成**

1. **Graphical Representation**

**图表

描述已自动生成**

The graphic shows the relationship between n & m is like linear, but I don’t think so, cause the deviation value is larger when the value of n is small. Then I used logarithm model and figured out this formula by trying a lot of times.

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

图形用户界面, 文本, 应用程序

描述已自动生成

电脑萤幕的截图

描述已自动生成