Chandrakanth Chittappa (001300076)

INFO 6205

Program Structures & Algorithms

Fall 2021

Assignment 1

Your task is

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 in terms of your observations and what you think might be going on.

NOTE: although I'm not going to tell you in advance what the relationship is, I can assure you that it is a simple relationship.

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

TASK 1

Graphical user interface, application, table, Excel

Description automatically generated

TASK 2

Task 2 is implemented in the main method of the UF\_HWQUPC.java class and the console output is given below:

Table

Description automatically generated

TASK 3 (Relationship between number of objects and number of pairs)

Chart, line chart

Description automatically generated

I have plotted the graph of sites/objects vs pairs required and objects vs objects. Since both are linear and similar, we can conclude that objects ∝ n