# Program Structures and Algorithms

## Spring 2023(SEC-01)

### Assignment 4

Name: Raja Shekar Reddy Siriganagari

NUID: 002762145

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

**Conclusion:**

Firstly, let us look at some values of the experiment performed over some values of number of objects(n) along with the number of random pairs generated to bring down the number of components down to 1 using union find.

|  |  |
| --- | --- |
| **n** | **m** |
| 250 | 762 |
| 500 | 1696 |
| 1000 | 3710 |
| 2000 | 8195 |
| 4000 | 17533 |
| 8000 | 38134 |
| 16000 | 81749 |
| 32000 | 175255 |
| 64000 | 369065 |
| 128000 | 708116 |
| 256000 | 1752646 |
| 512000 | 3236582 |
| 1024000 | 7098253 |
| 2048000 | 15377697 |
| 4096000 | 32351636 |
| 8192000 | 69786548 |

The relationship between the number of objects (n) and the number of random pairs(m) reduce the number of components to 1 using union find is:

There are n objects in total on whom the union operation is called at-most n-1 times. The graph plotted for n and m is shown below.

Chart, line chart

Description automatically generated

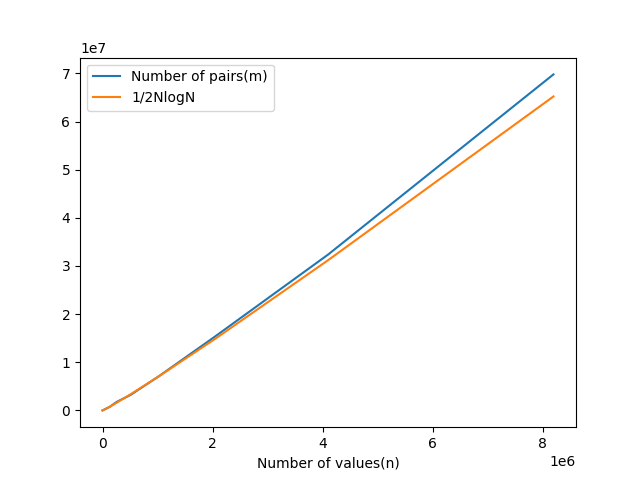
**Evidence to support this conclusion:**

The relationship can be proved by referring to the Erdős–Rényi model where they have concluded that in a model where there is an equal probability to choose a random edge the probability of the graph being connected after approximately operations is 50-50

With that equation in mind let us generate the values for our existing values and try to plot them.

|  |  |  |
| --- | --- | --- |
| **n** | **m** | **½ n\*ln n** |
| 250 | 762 | 690 |
| 500 | 1696 | 1554 |
| 1000 | 3710 | 3454 |
| 2000 | 8195 | 7601 |
| 4000 | 17533 | 16588 |
| 8000 | 38134 | 35949 |
| 16000 | 81749 | 77443 |
| 32000 | 175255 | 165976 |
| 64000 | 369065 | 354132 |
| 128000 | 708116 | 752626 |
| 256000 | 1752646 | 1593975 |
| 512000 | 3236582 | 3365396 |
| 1024000 | 7098253 | 7085684 |
| 2048000 | 15377697 | 14881151 |
| 4096000 | 32351636 | 31181868 |
| 8192000 | 69786548 | 65202867 |

Plotting a graph for these three columns will show us that the equation is approximately accurate.



**Unit test screenshots:**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**