

Lan Gao(001568670)

Program Structures & Algorithms Fall 2021

Assignment No.3 Union-find

- Task

Part1

Implementation

```
public int find(int p) {
    validate(p);
    int root = p;
    // TO BE IMPLEMENTED
    while(root !=getParent(root)){
        if (this.pathCompression) {
            doPathCompression(root);
        }
        root = getParent(root);
    }
    return root;
}
```

```
private void mergeComponents(int i, int j) {
    // TO BE IMPLEMENTED make shorter root point to taller one
    if (height[i]<height[j]){
        updateParent(i,j);
        updateHeight(j,i);
    }else{
        updateParent(j,i);
        updateHeight(i,j);
    }
}

/**
 * This implements the single-pass path-halving mechanism of path compression
 */
private void doPathCompression(int i) {
    // TO BE IMPLEMENTED update parent to value of grandparent
    updateParent(i,parent[parent[i]]);
}
```

Part2

Implementation

```
public static int count(int n, boolean doPathCompression){
    UF_HWQUPC uf = new UF_HWQUPC(n, doPathCompression);
    Random random = new Random();
    boolean generated[] = new boolean[n];
    Arrays.fill(generated, false);
    int connections=0;
    boolean loop=true;
    while(loop) {
        int p = random.nextInt(n);
        int q = random.nextInt(n);
        generated[p] = true;
        generated[q] = true;
        connections++;
        if (!uf.connected(p, q)) {
            uf.union(p, q);
        }
        int index = 0;
        for (;index < n ; index++) {
            if(!generated[index]) break;
        }
        if(index==n)loop=false;
    }
    return connections;
}

public static void main(String[] args) {
    int n = 150;
    int totalCount = 0;
    for (int i = 0; i <100 ; i++) {
        int connectNum=count(n, doPathCompression: true);
        totalCount+=connectNum;
        System.out.println("The number of sites is: "+n+"\nThe number of connections is: "+connectNum+"\n");
    }
    System.out.println("The average number of connections is: "+totalCount/100.0);
}
```

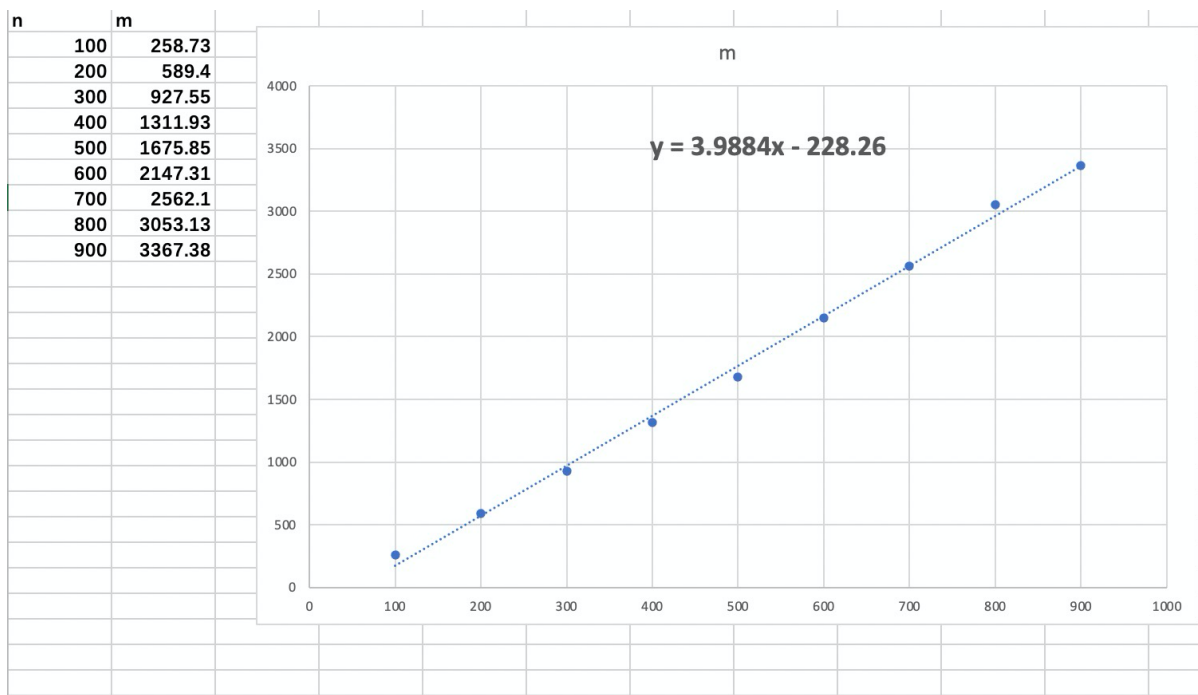
- **Relationship Conclusion**

The relationship between the number of objects (n) and the number of pairs (m) can be

$$M = 3.9884N - 228.261$$

This could be proved from the output below.

Result



Output

Take 1 n for example

- n=900

```
Run: UF_HWQUPC x
The number of sites is: 900
The number of connections is: 3096

The number of sites is: 900
The number of connections is: 3619

The number of sites is: 900
The number of connections is: 2848

The number of sites is: 900
The number of connections is: 2970

The number of sites is: 900
The number of connections is: 2972

The number of sites is: 900
The number of connections is: 4248

The average number of connections is: 3367.38

Process finished with exit code 0
```

- Unit tests result

```
Run: UF_HWQUPC_Test x
Tests passed: 13 of 13 tests - 11 ms
UF_HWQUPC_Test (edu.neu.coe.info6205.union.fi 11 ms)
  ✓ testIsConnected01 8 ms
  ✓ testIsConnected02 0 ms
  ✓ testIsConnected03 2 ms
  ✓ testFind0 0 ms
  ✓ testFind1 0 ms
  ✓ testFind2 0 ms
  ✓ testFind3 1 ms
  ✓ testFind4 0 ms
  ✓ testFind5 0 ms
  ✓ testToString 0 ms
  ✓ testConnect01 0 ms
  ✓ testConnect02 0 ms
  ✓ testConnected01 0 ms
Process finished with exit code 0
```

