PA4 Report

Junxi Li, Lingfei Lu

We use two separate test cases:

i) 100 same pairs
ii) 100 different pairs

For the first one, bfs used 7.08 seconds while union-find used 2.29 seconds.
For the second one, bfs used 24.2 seconds and union-find used 2.13 seconds.
Analyze: We first connect all the actors, and connect every actor with edges; so we did not follow what the write-up told us to do: load all actors first and add movies sequentially. And when we do BFS, we use bfs, we only search those edge which contains a movie whose year is less than or equal to a given year. This year is given by the earlier year between two actors started to act a movie. And if we did not find the connection this year, we increment the year until the last year, say 2015/2016 (not sure).
Therefore, for bfs, we have to run approximately O(V + E) (although not all the edges), and we have to run up to about 30 times in average.
For Union-Find, we have approximately O(log N), for N is number of connected nodes in the graph. Since this is a tree structure, and although it has a path compression, it won't matter that much.

1. Union-find is better since union-find algorithm performed better in both cases and for the first case, it is faster than bfs 3 times, and second case about 12 times.
2. When the test pairs are different and probably some test cases need more bfs, then union-find would out-perform bfs.
3. For the first one, bfs used 7.08 seconds while union-find used 2.29 seconds.
   For the second one, bfs used 24.2 seconds and union-find used 2.13 seconds.