Robert Morain

1. [10 points] In the PDF report, explain the time complexity of both implementations of the algorithm by showing and summing up the complexity of each subsection of your code.

Heap implementation

Get_next= O(log(V))
Update_key = O(V)
Dijkstra runs V times
Update key runs O(V^2) at the worst case

In total O(V^3) because of the update key
Bad for highly connected components

Array implementation

Get_next= O(V)
Update_key = O(1)
Dijkstra runs V times
Update key runs O(V^2) at the worst case
Get_next is run V times

In total O(V^2) because of the update key

[20 points] For Random Seed 42 – Size 20, Random Seed 123 – Size 200 and Random Seed 312 – Size 500: include a screenshot in your PDF report showing the shortest path (if one exists) for each of the three source–destination pairs (see the images included at the bottom of this document).

- For Random seed 42 – Size 20, use node 7 (the left–most node) as the source and node 1 (on the bottom toward the right) as the destination, as in the first image below.
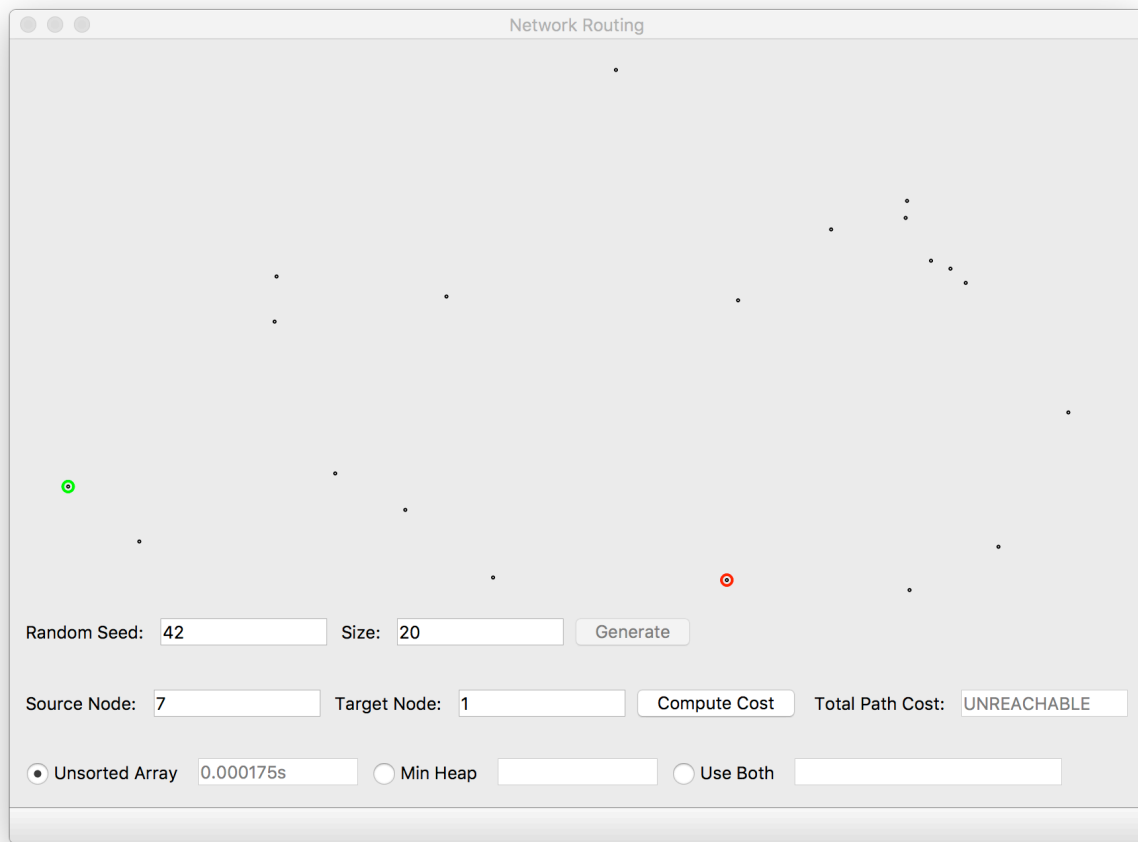
- For Random seed 123 – Size 200, use node 94 (near the upper left) as the source and node 3 (near the lower right) as the destination, as in the second image below.
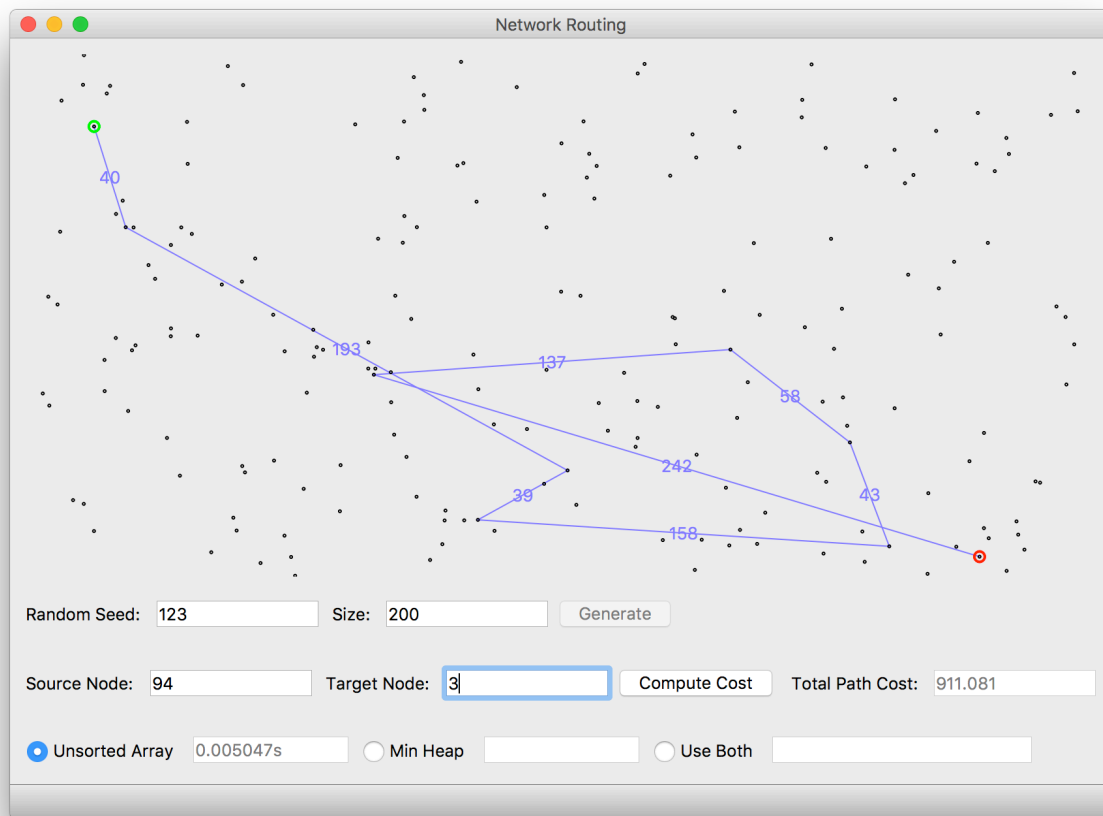
- For Random seed 312 – Size 500, use node 2 (near the lower left) as the source and node 8 (near the upper right) as the destination, as in the third image below.
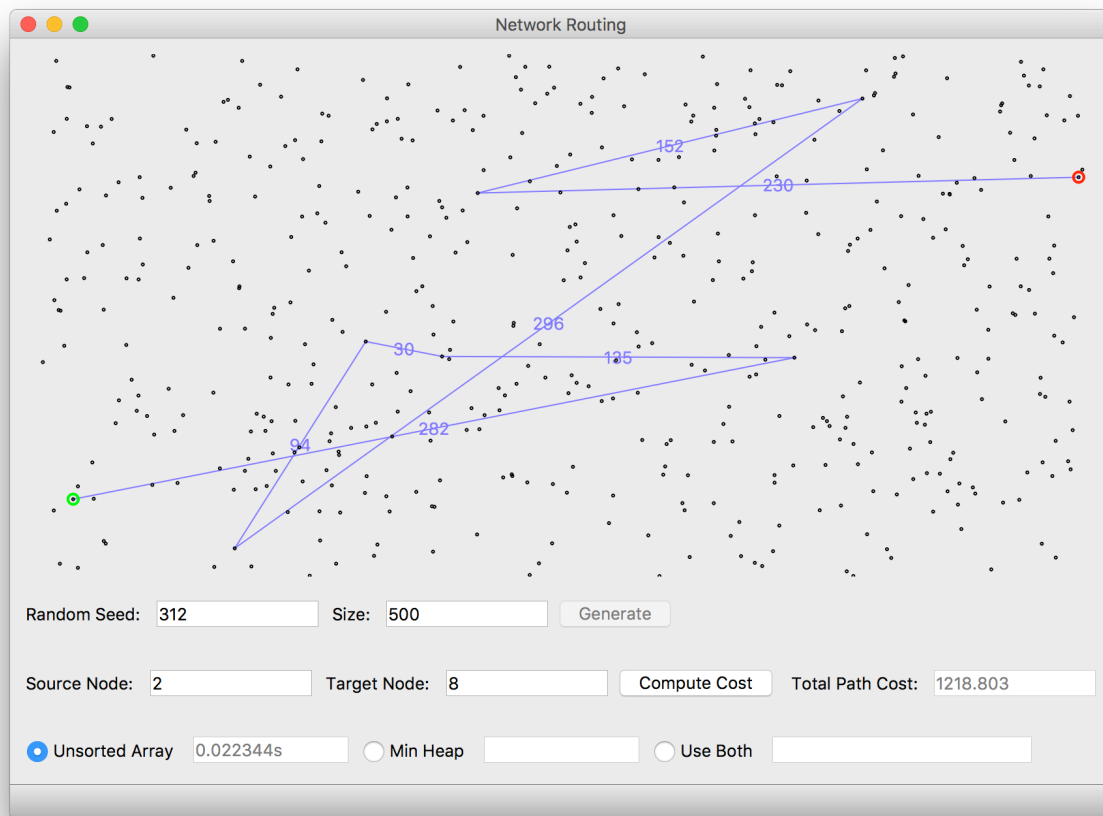
1. **[20 points]** For different numbers of nodes (100, 1000, 10000, 100000, 1000000), compare the empirical time complexity for Array vs. Heap. Give your best estimate of the difference (for 1000000 nodes, run only the heap version and then estimate how long you might expect your array version to run based on your other results). For each number of nodes do at least five tests with different random seeds, and average the results. Redo any case where the destination is unreachable. Each time, start with nodes approximately in opposite corners of the network. In your PDF report, include a graph of your results and also give a table of your raw data (data for each of the runs); in both graph and table, include your one estimated runtime (array implementation for 1000000 points). Discuss the results and give your best explanations of why they turned out as they did.

Size	Array	Heap	Heap is how many times faster?
100	.001558s	.002865s	.554
1000	.074218s	.070748	1.049
10000	5.806775s	5.566004s	1.043
100000			
1000000			