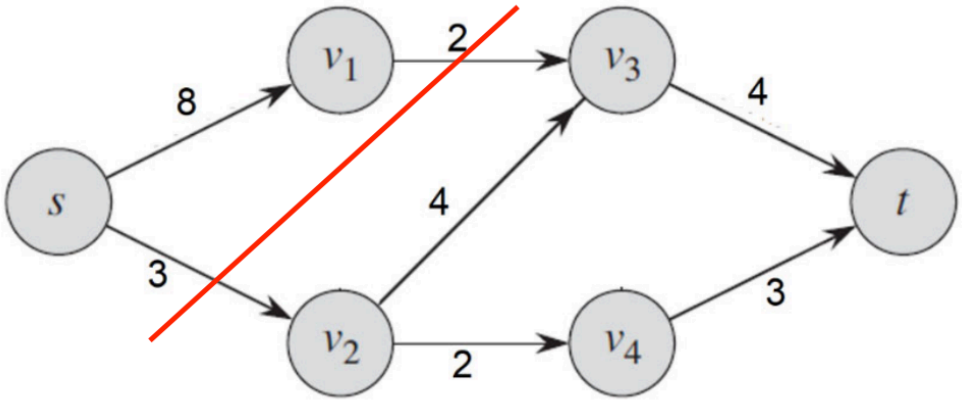
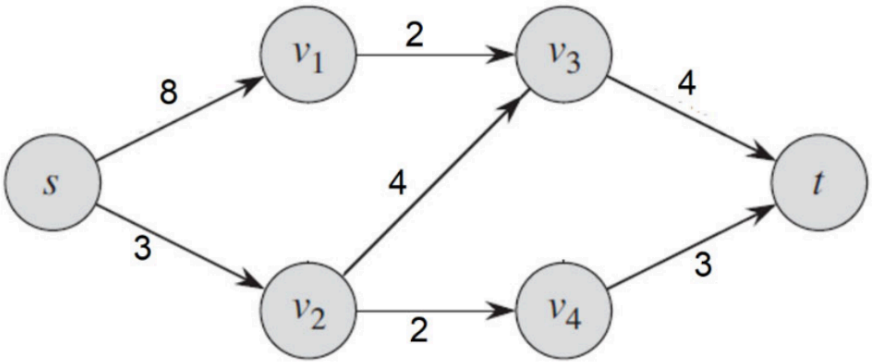
Alex Sadler

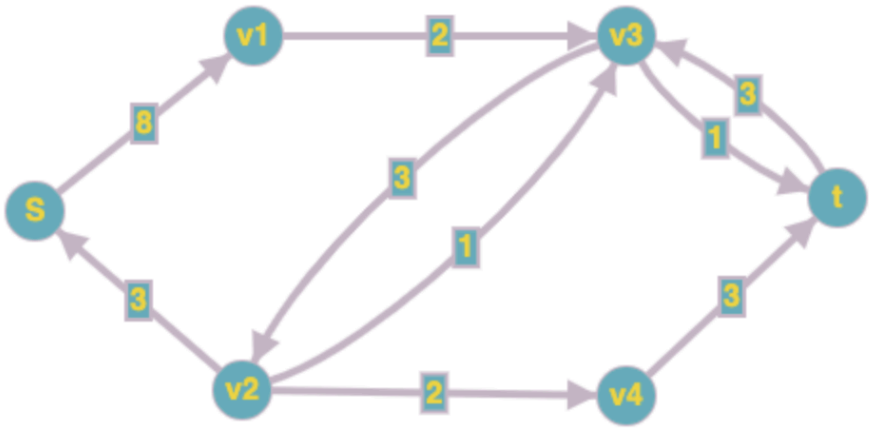
Problem 1

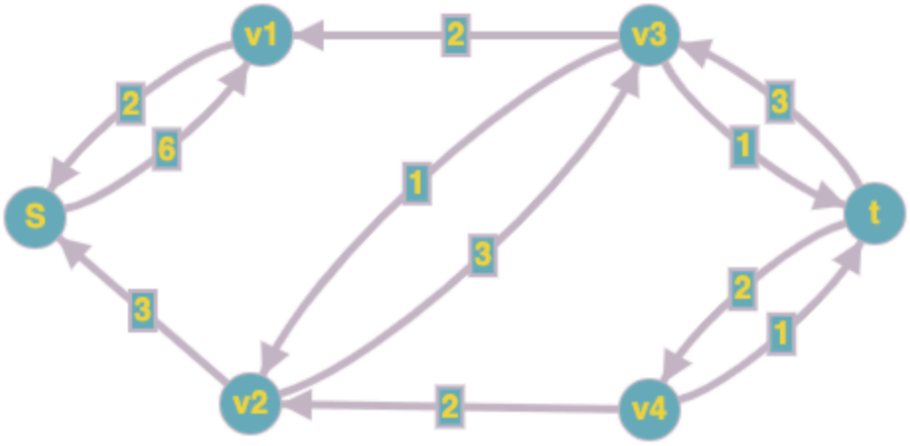
- Minimum cut: 5



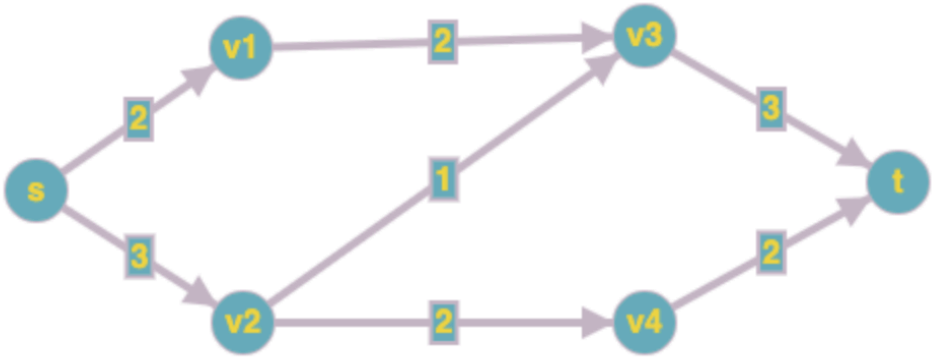
- Applying algorithm and showing residual networks:







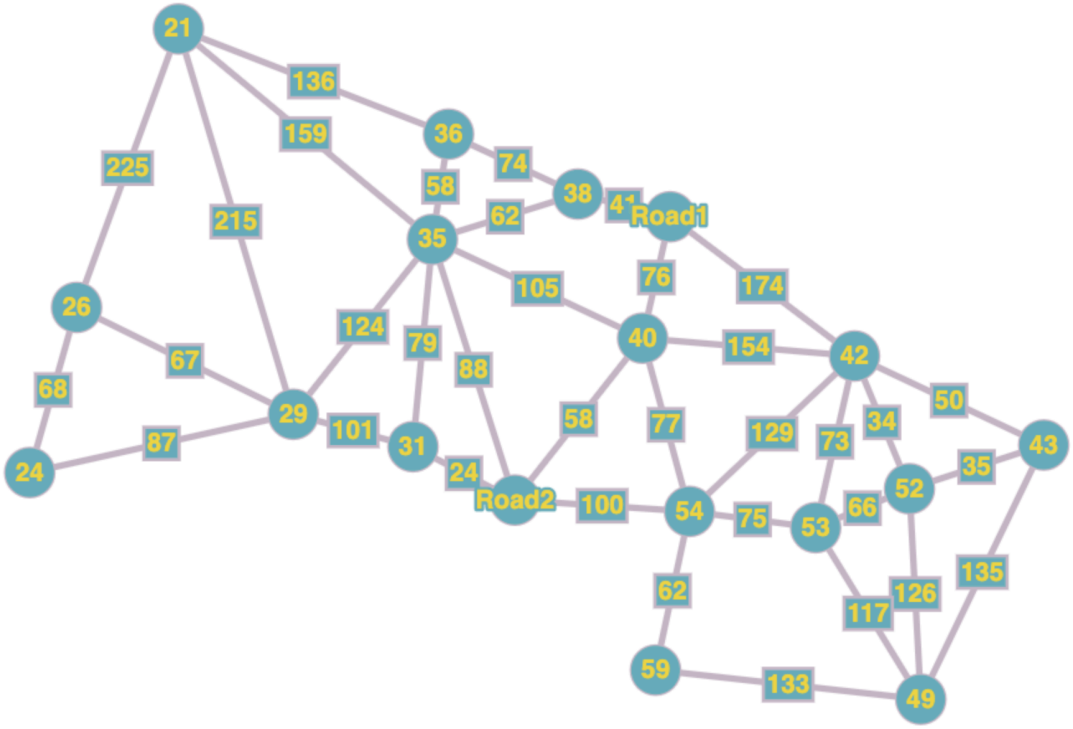
Now there are no more augmenting paths and the flow matches the maximum flow of the minimum cut (5), so we have the final flow network.



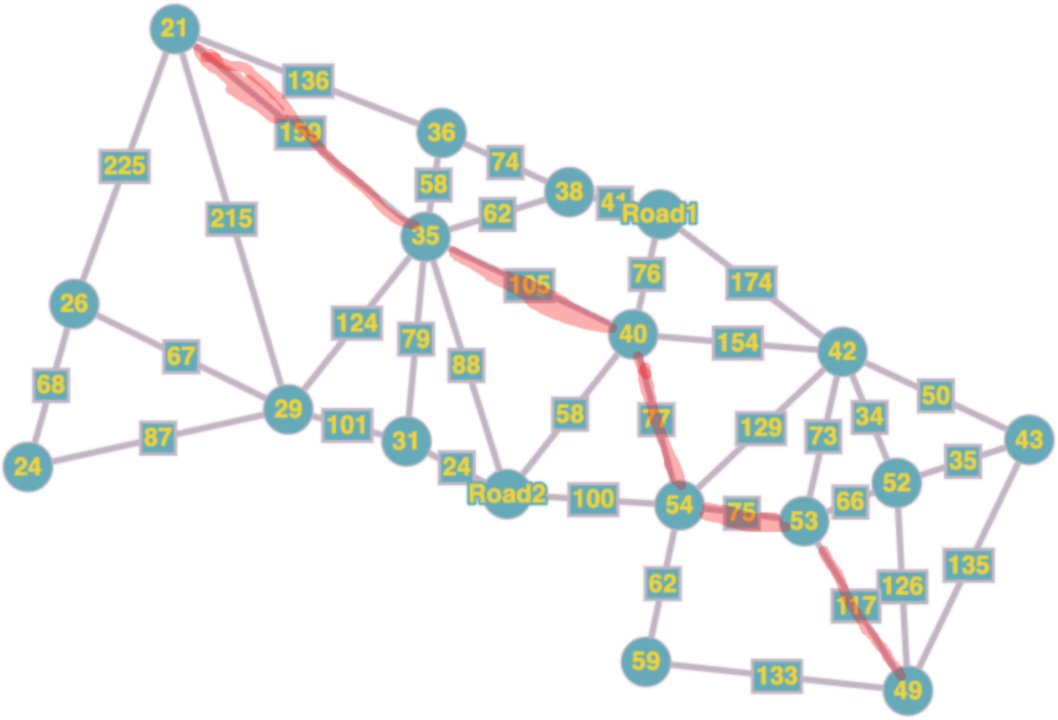
Alex Sadler

Problem 2

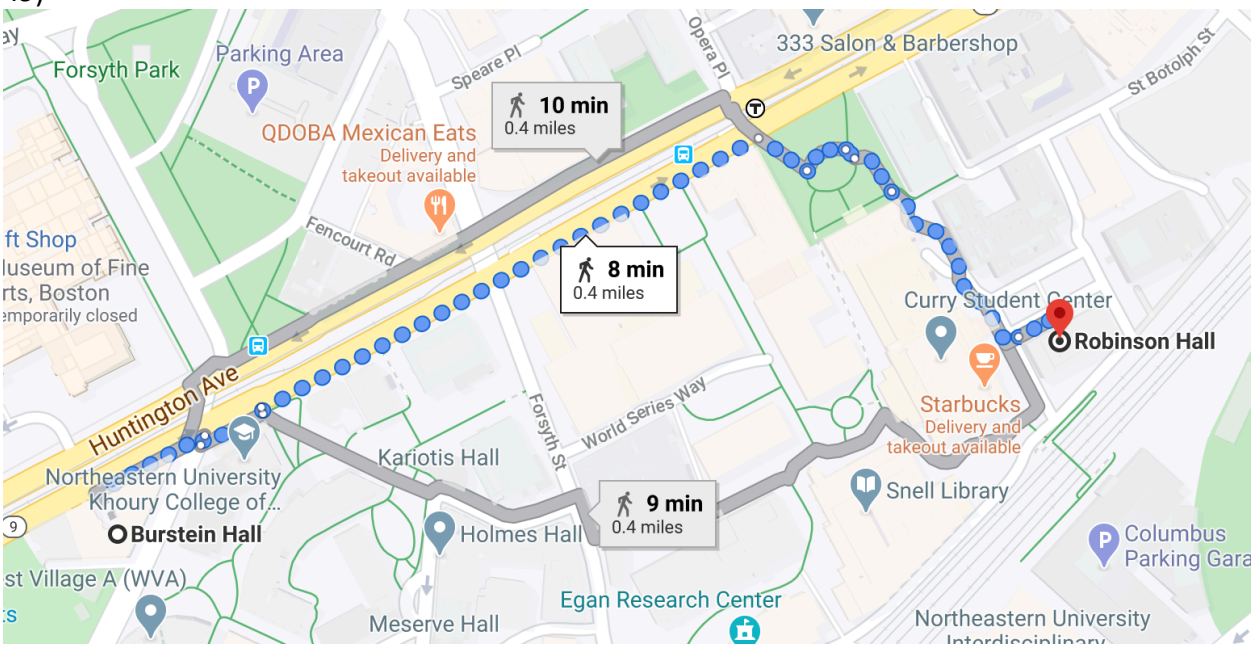
Undirected graph of campus map (weights are meters between buildings/intersections) is below. I've denoted Road1 as "building" 1 and Road2 as "building" 2 in my adjacency matrix.



Sample run 1 from node 21 to 49:



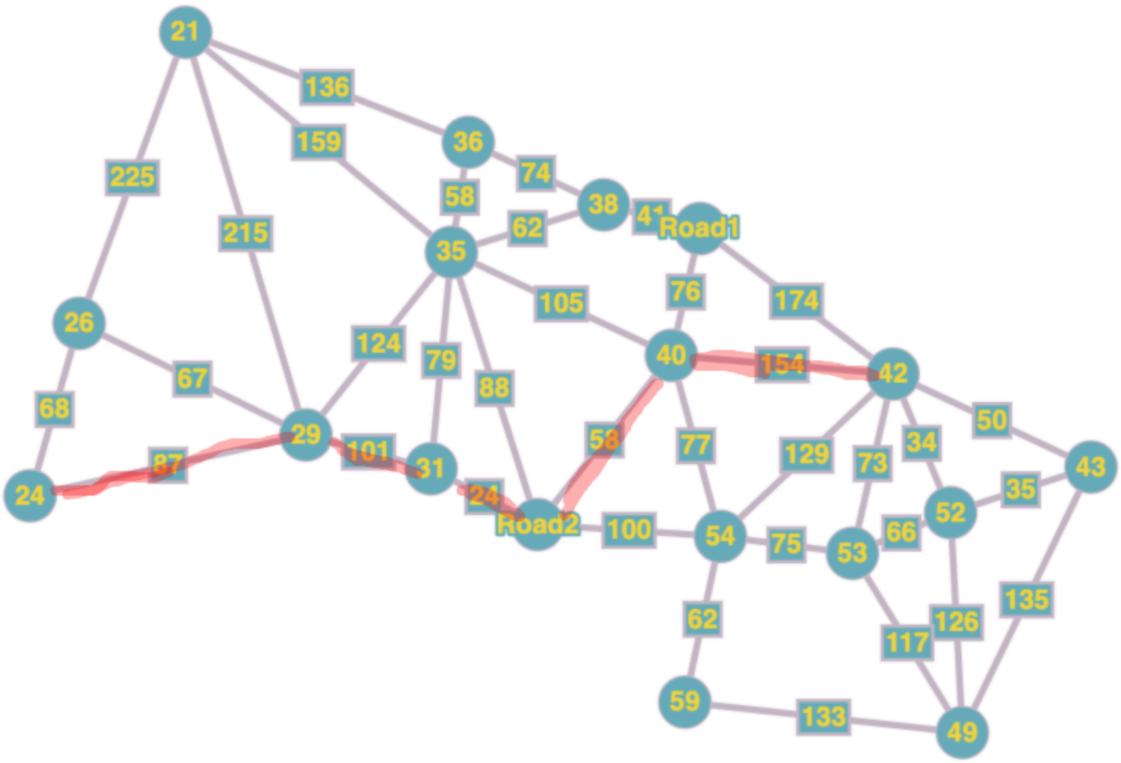
- Path: 21 → 35 → 40 → 54 → 53 → 49
- Total distance: 533 meters
- Google maps walking results from Burnstein Hall (building 21) to Robinson Hall (building 49)



  - The fastest walking route according to Google Maps is along Huntington Avenue, but the second fastest result is similar to the path my code output with a distance of 0.4 miles (roughly 640 meters)
- Output:

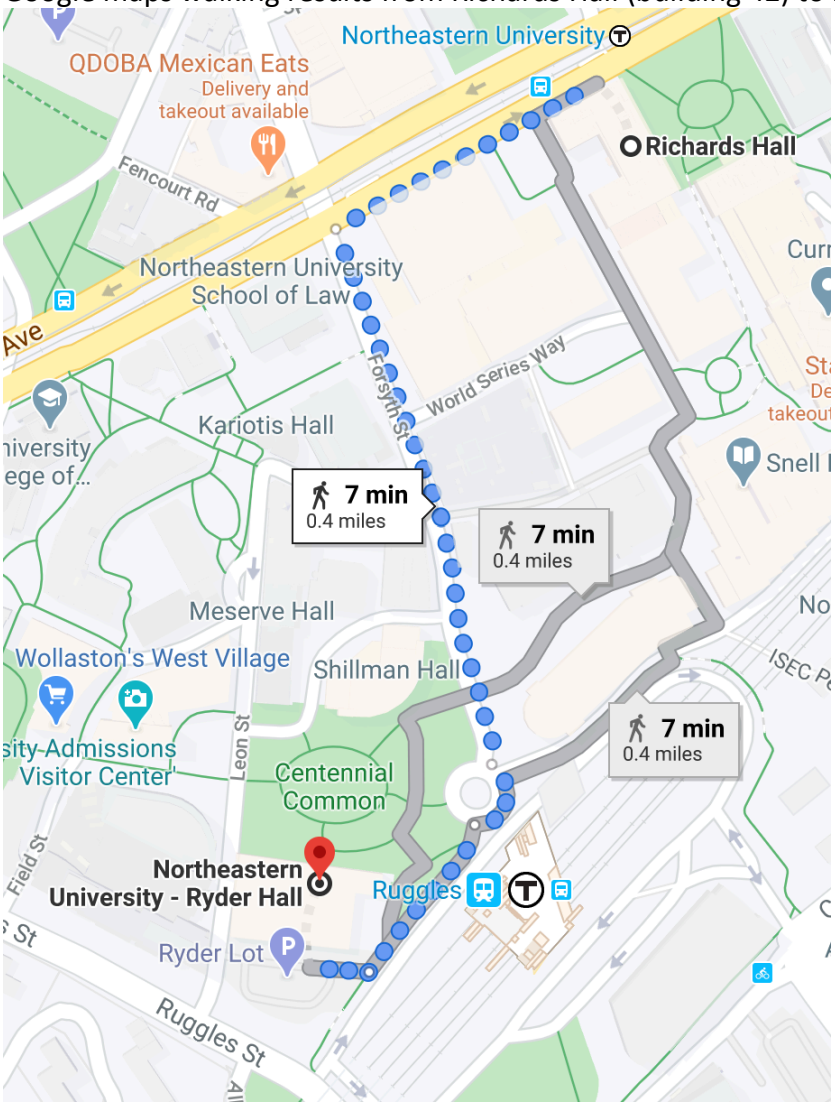
```
[-bash-4.2$ g++ -std=c++11 main.cpp
[-bash-4.2$ ./a.out
[Enter start vertex: 21
[Enter end vertex: 49


Distance from building 21 to building 1: 251. Predecessor: 38
Distance from building 21 to building 2: 247. Predecessor: 35
Distance from building 21 to building 21: 0. Predecessor: -1
Distance from building 21 to building 24: 293. Predecessor: 26
Distance from building 21 to building 26: 225. Predecessor: 21
Distance from building 21 to building 29: 215. Predecessor: 21
Distance from building 21 to building 31: 238. Predecessor: 35
Distance from building 21 to building 35: 159. Predecessor: 21
Distance from building 21 to building 36: 136. Predecessor: 21
Distance from building 21 to building 38: 210. Predecessor: 36
Distance from building 21 to building 40: 264. Predecessor: 35
Distance from building 21 to building 42: 418. Predecessor: 40
Distance from building 21 to building 43: 468. Predecessor: 42
Distance from building 21 to building 49: 533. Predecessor: 53
Distance from building 21 to building 52: 452. Predecessor: 42
Distance from building 21 to building 53: 416. Predecessor: 54
Distance from building 21 to building 54: 341. Predecessor: 40
Distance from building 21 to building 59: 403. Predecessor: 54

[Path from building 49 to building 21: 49, 53, 54, 40, 35, 21-bash-4.2$
-bash-4.2$
```

Sample run 2 from node 42 to 24:

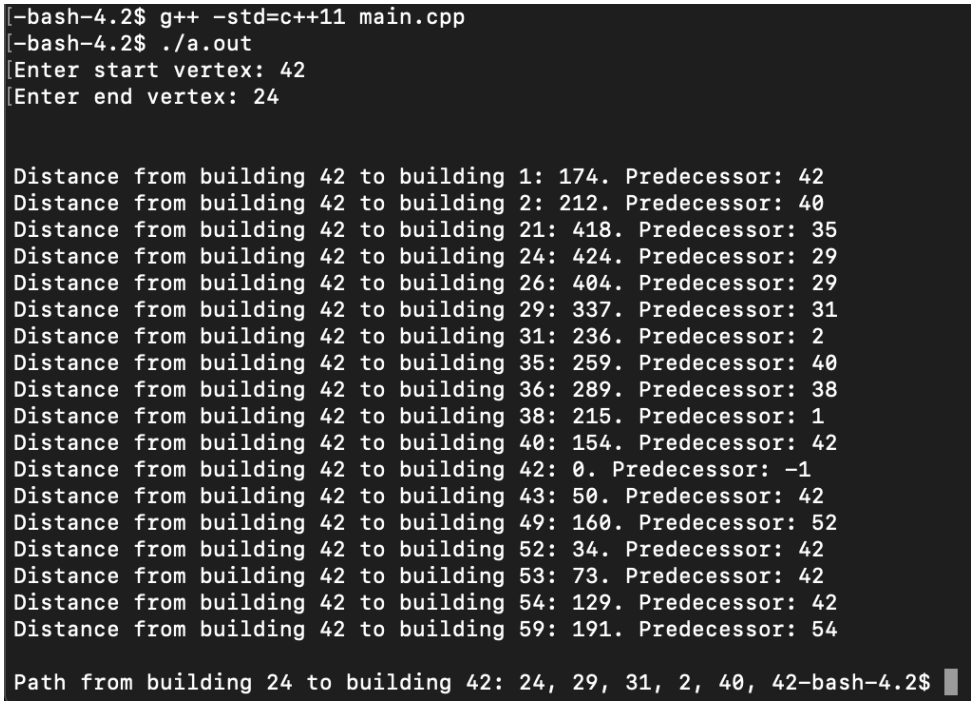


- Path: 42 → 40 → Road2 (an intersection) → 31 → 29 → 24
- Total distance: 424 meters

- Google maps walking results from Richards Hall (building 42) to Ryder Hall (building 24)



  - The fastest walking route according to Google Maps is very similar to the path my code output with a distance of 0.4 miles, which is roughly 640 meters.

- Output:
  - 
-bash-4.2$ g++ -std=c++11 main.cpp
-bash-4.2$ ./a.out
Enter start vertex: 42
Enter end vertex: 24
Distance from building 42 to building 1: 174. Predecessor: 42
Distance from building 42 to building 2: 212. Predecessor: 40
Distance from building 42 to building 21: 418. Predecessor: 35
Distance from building 42 to building 24: 424. Predecessor: 29
Distance from building 42 to building 26: 404. Predecessor: 29
Distance from building 42 to building 29: 337. Predecessor: 31
Distance from building 42 to building 31: 236. Predecessor: 2
Distance from building 42 to building 35: 259. Predecessor: 40
Distance from building 42 to building 36: 289. Predecessor: 38
Distance from building 42 to building 38: 215. Predecessor: 1
Distance from building 42 to building 40: 154. Predecessor: 42
Distance from building 42 to building 42: 0. Predecessor: -1
Distance from building 42 to building 43: 50. Predecessor: 42
Distance from building 42 to building 49: 160. Predecessor: 52
Distance from building 42 to building 52: 34. Predecessor: 42
Distance from building 42 to building 53: 73. Predecessor: 42
Distance from building 42 to building 54: 129. Predecessor: 42
Distance from building 42 to building 59: 191. Predecessor: 54
Path from building 24 to building 42: 24, 29, 31, 2, 40, 42-bash-4.2$