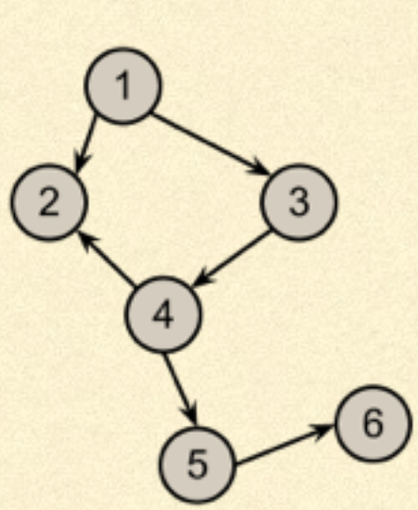
DIJKSTRA'S ALGORITHM: FINDING THE MOST EFFICIENT PATH TO CLASS

Ethan Stanley
CS 39

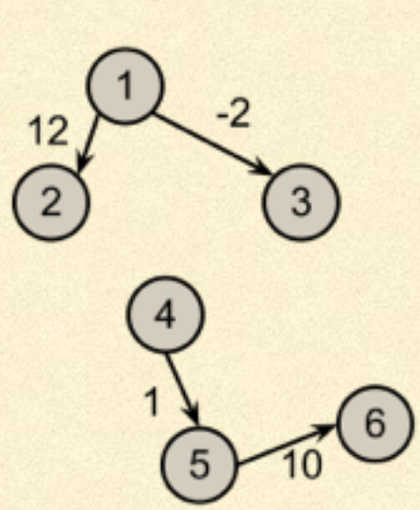
GRAPH BASICS

- A graph is a grouping of vertices(nodes) connected by edges
- Graphs can be connected or disconnected
- Connected graph: there is a path between any two given vertices in the graph
- Disconnected graph: not all vertices in graph are connected
- For Dijkstra's we are mainly concerned with connected graphs

Connected Graph



Connected



Disconnected

GRAPH BASICS CONTINUED

- Graphs can be directed and undirected
- Directed graphs: The edges of the graph point in one direction (such as a one way street)
- Undirected graphs: The edges of the graph have no direction (such as a two way street)
- For Dijkstra's Algorithm we can work with either directed or undirected graphs

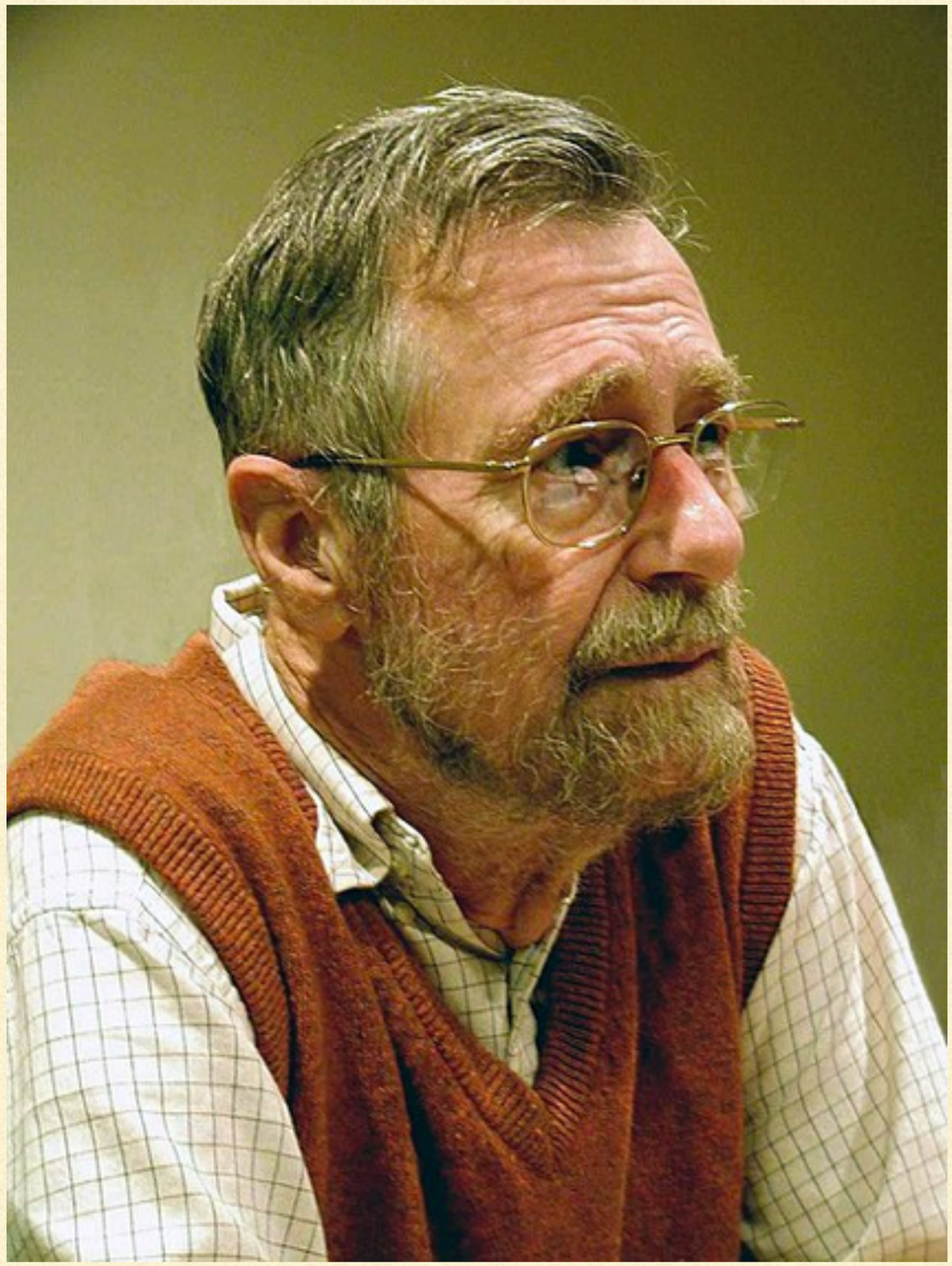
GRAPH BASICS CONTINUED

- Graphs can be weighted or unweighted
- Weighted graph: the edges of the graph have weights (as in the length of a street)
- Unweighted graph: the edges of the graph do not have weights (as if all streets were of same length)
- For Dijkstra's Algorithm we will only concern ourselves with weighted graphs

Dijkstra's Algorithm Introduction

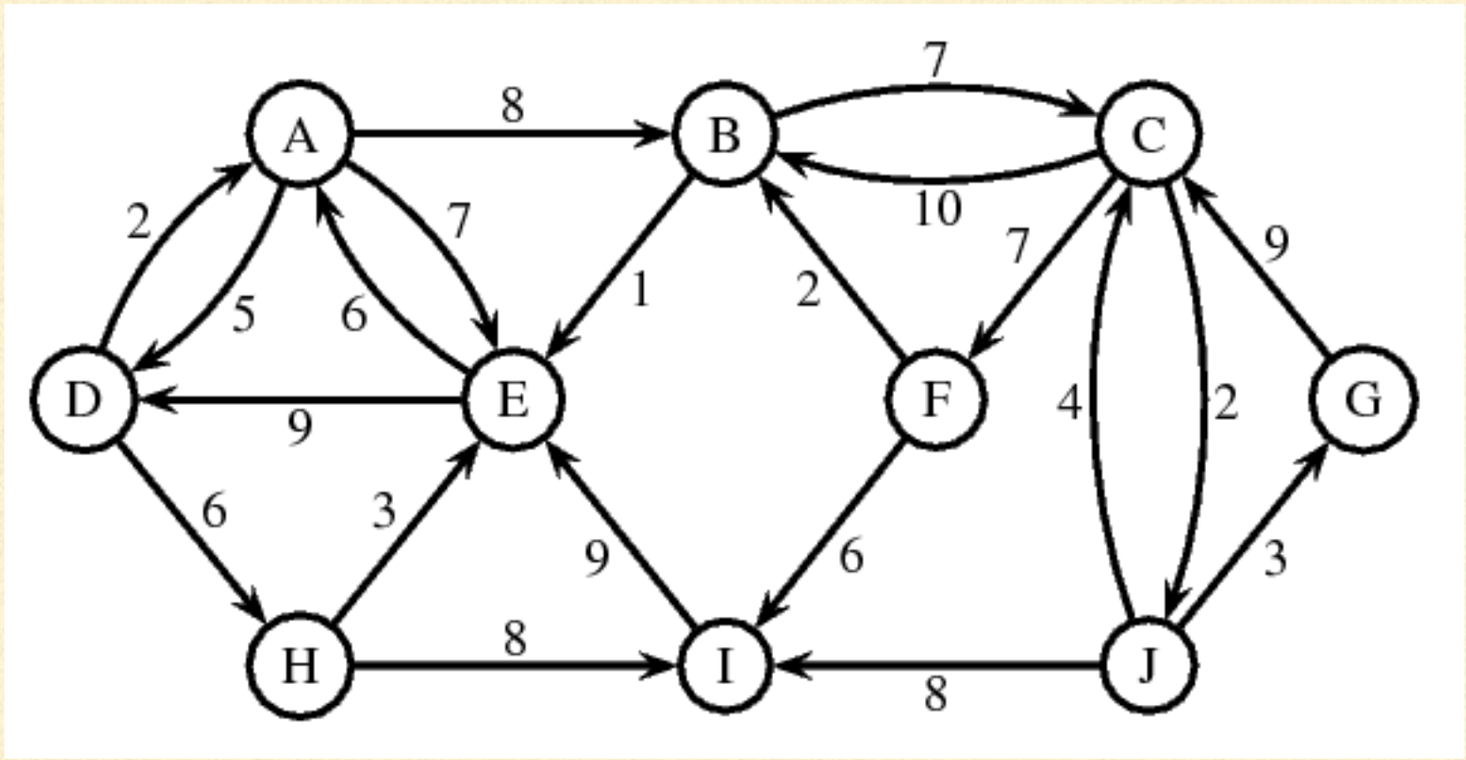
- Dijkstra's was conceived in in 1956 by computer scientist Edsger Wybe Dijkstra
- The most common variant of the algorithm fixes a single node as the source node and finds shortest paths from the source to all other nodes
- Path finding algorithms are very important, and we use them all the time in applications such as google maps and other navigation systems.
- This application will be demonstrated in the following slides. Think of edges as roads or paths that you must use to travel to a certain location. Your goal is to find a combination of roads that yield the shortest total distance to your destination.

https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm



Edsger Wybe Dijkstra

A DIJKSTRA'S WORTHY GRAPH



Directed (check) Weighted (check) Connected (check)

https://www.cs.usfca.edu/~galles/visualization/Dijkstra.html

DIJKSTRA'S

Dijkstra's Demo

- Assume you have a complete graph with x nodes
- Step 1: Pick a start node, and mark all other nodes as unvisited (Dijkstra's algorithm will find the shortest path from this node to every other node in this graph)
- Step 2: Look at all the nodes adjacent to the node you are on and examine their distances (adjacent nodes are connected to your start node with one edge)

  A. Distance to node (1): 2 B. Distance to node (2): 1
- Step 3: Visit the closest node, and mark it as visited (it will not be checked again)

  A. Visit node 2
- Step 4: Visit the next closest node that has not yet been visited (only look at nodes adjacent to visited nodes)

  A. Distance to node (1): 2 B. Distance to node (5): 16 C. Visit Node 1
- Step 5: Visit the next closest node that has not yet been visited (only look at nodes adjacent to visited nodes)

  A. Distance to node (3): 13 B. Distance to node (4): 5 C. Distance to node (5): 16 D. Visit node (4)

- Step 6: Visit the next closest node that has not yet been visited (only look at nodes adjacent to visited nodes)

  A. Distance to node (5): 9 B. Distance to node (6): 10 C. Distance to node (3): 13 D. Visit node (5)

- Step 7: Compare the last two nodes

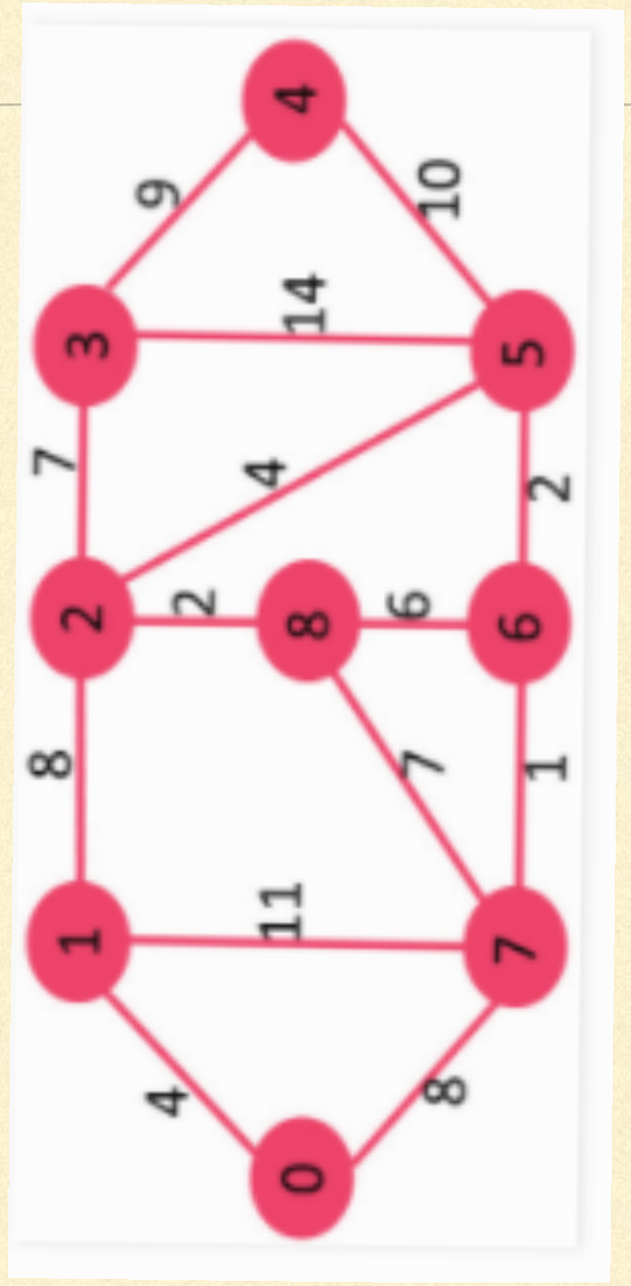
  A. Distance to node (6): 10 B. Distance to node (3): 13 C. Visit node (6)

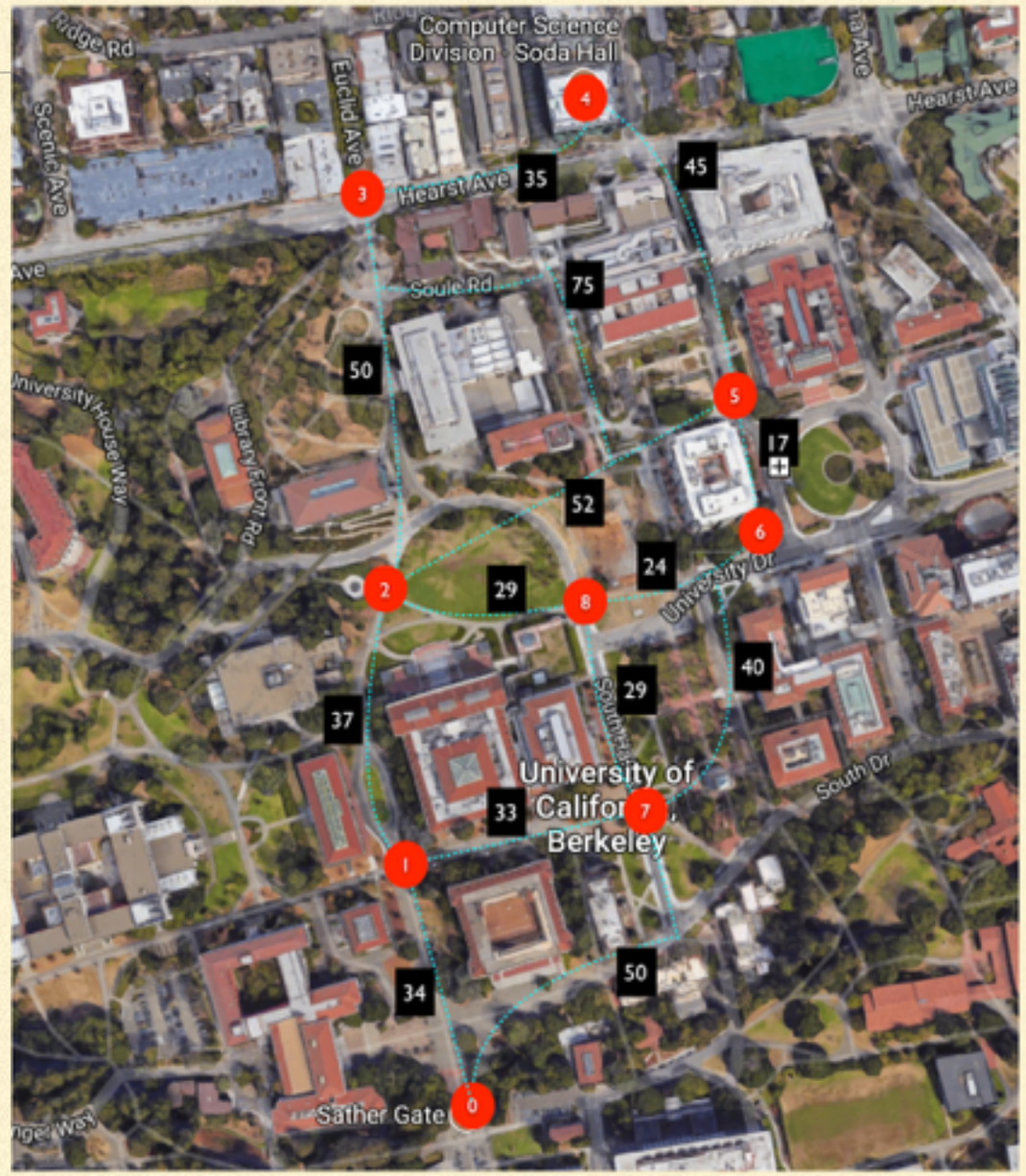
- Step 8: Visit the last remaining node

  A. Distance to node (3) is now: 11 B. Visit node (3)

NODE #	0	1	2	3	4	5	6
DISTANCE	0	2	1	13	5	9	10
NODES USED	0	0,1	0,2	0	0,1,4,6,3	0,1,4,5	0,1,4,6

APPLYING DIJKSTRA'S ALGORITHM TO CAMPUS



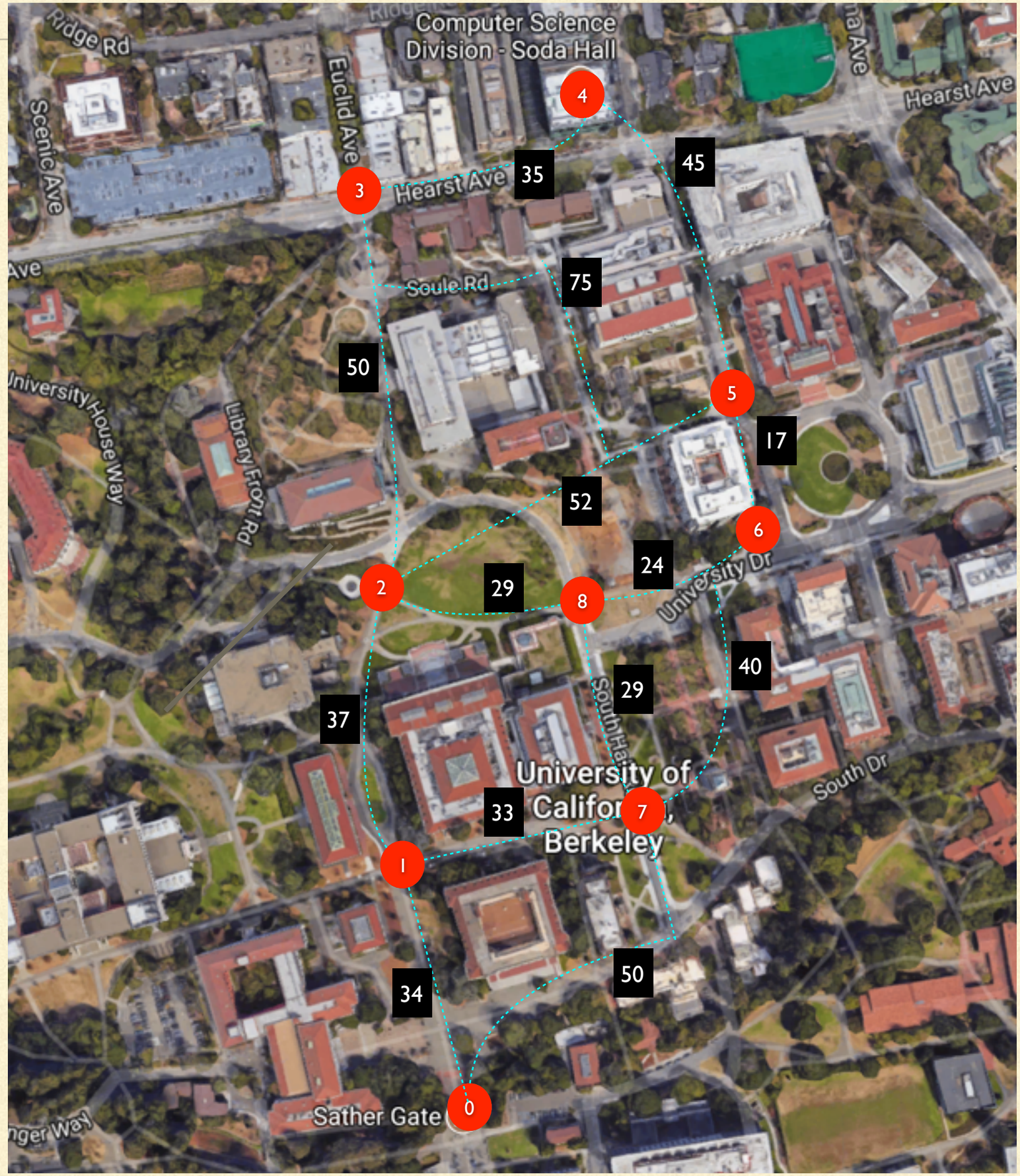


http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/

- Graph Characteristics: connected, weighted, and undirected.

0-7 : 50
7-6 : 40 +
6-5 : 17 +
5-4 : 45 +

152



http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/

The End