#40: Graph Puzzles + End of Semester

April 27, 2018 · *Wade Fagen-Ulmschneider*

Dijkstra’s Algorithm Overview:

- The overall logic is the same as Prim’s Algorithm
- We will modify the code in only two places – both involving the update to the distance metric.
- The result is a directed acyclic graph or DAG containing the shortest path to every vertex from a single starting point.

Pseudocode for Dijkstra’s SSSP Algorithm

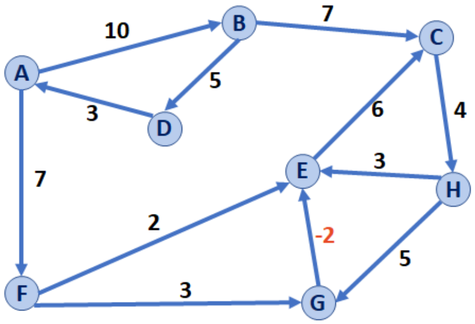
```
DijkstraSSSP(G, s):
  foreach (Vertex v : G):
    d[v] = +inf
    p[v] = NULL
  d[s] = 0

  PriorityQueue Q   // min distance, defined by d[v]
  Q.buildHeap(G.vertices())
  Graph T           // "labeled set"

  repeat n times:
    Vertex m = Q.removeMin()
    T.add(m)
    foreach (Vertex v : neighbors of m not in T):
      if _______________________ < d[v]:
        d[v] = _________________________
        p[v] = m

  return T
```

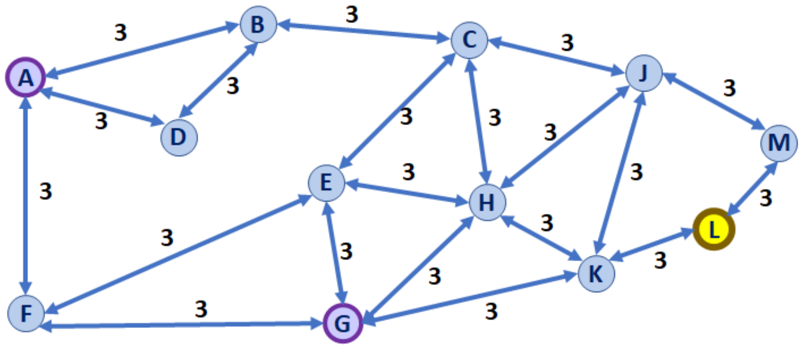
Dijkstra: What if we have a minimum-weight edge, without having a negative-weight cycle?



Dijkstra’s Algorithm optimality assumption:

Dijkstra’s Algorithm running time:

Challenge #1: Landmark Path Problem



End of Semester Logistics

CS 225 Final Exam

- The final exam begins on Thursday, May 3rd
- The final exam is a 3 hour CBTF exam, is a cumulative exam, and has the format of a combined theory + programming exam
- The last office hours is Wednesday, May 2nd
- *We'll use lecture on Wednesday, May 2nd as a final exam review!*

"Pre-Final" Grade Dump

- I believe there's only a few remaining issues left with grading; I'll be starting to wrap these up myself over the weekend:
    - +EC from creative components
    - Working on recovering repos that were force deleted
- As soon as possible after MP7's deadline, we'll provide a "Pre-Final" grade in Compass that incorporates everything except the final exam into your CS 225 grade.

End of Semester Grade Review

- Excel sheet will be provided once final grades are posted.
- Must submit an Excel sheet for this review.

Floyd-Warshall Algorithm

Floyd-Warshall's Algorithm is an alternative to Dijkstra in the presence of negative-weight edges (but <u>not</u> negative weight cycles).

Pseudocode for Floyd-Warshall's Algorithm

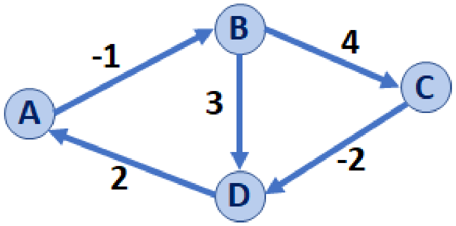
```
FloydWarshall(G):
  Input: G, Graph;
  Output: d, an adjacency matrix of distances between
all
          vertex pairs

  Let d be a adj. matrix initialized to +inf
  foreach (Vertex v : G):
    d[v][v] = 0
  foreach (Edge (u, v) : G):
    d[u][v] = cost(u, v)

  foreach (Vertex u : G):
    foreach (Vertex v : G):
      foreach (Vertex w : G):
        if d[u, v] > d[u, w] + d[w, v]:
          d[u, v] = d[u, w] + d[w, v]

  return d
```

Running Floyd-Warshall:



	A	B	C	D
A				
B				
C				
D				

CS 225 – Things To Be Doing:

1. Final Exam runs Thursday, May 3 – Thursday, May 10
2. MP7 deadline Monday, April 30
3. Final lab, **lab_ml** due Sunday, April 29
4. Final POTD is right now! ☹