Dec. 6 – Dijkstra's algorithm
Wade Fagen-Ulmschneider
Prim's Algorithm

```
PrimMST(G, s):
6    foreach (Vertex v : G):
7        d[v] = +inf
8        p[v] = NULL
9        d[s] = 0
10
11    PriorityQueue Q // min distance, defined by d[v]
12    Q.buildHeap(G.vertices())
13    Graph T         // "labeled set"
14
15    repeat n times:
16        Vertex m = Q.removeMin()
17        T.add(m)
18        foreach (Vertex v : neighbors of m not in T):
19            if cost(v, m) < d[v]:
20                d[v] = cost(v, m)
21                p[v] = m
```

<table>
<thead>
<tr>
<th>Adj. Matrix</th>
<th>Adj. List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heap</td>
<td></td>
</tr>
<tr>
<td>Unsorted Array</td>
<td></td>
</tr>
</tbody>
</table>

Prim’s Algorithm

Sparse Graph:

Dense Graph:

<table>
<thead>
<tr>
<th>Adj. Matrix</th>
<th>Adj. List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heap</td>
<td>$O(n^2 + m \log(n))$</td>
</tr>
<tr>
<td>Unsorted Array</td>
<td>$O(n^2)$</td>
</tr>
</tbody>
</table>
MST Algorithm Runtime:

- Kruskal’s Algorithm: $O(n + m \log(n))$
- Prim’s Algorithm: $O(n \log(n) + m \log(n))$

• What must be true about the connectivity of a graph when running an MST algorithm?

• How does $n$ and $m$ relate?
MST Algorithm Runtime:

• Upper bound on MST Algorithm Runtime: $O(m \ lg(n))$
Suppose I have a new heap:

```java
PrimMST(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 cost(v, m) < d[v]:
            d[v] = cost(v, m)
            p[v] = m
```

<table>
<thead>
<tr>
<th></th>
<th>Binary Heap</th>
<th>Fibonacci Heap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remove Min</td>
<td>O( lg(n) )</td>
<td>O( lg(n) )</td>
</tr>
<tr>
<td>Decrease Key</td>
<td>O( lg(n) )</td>
<td>O(1)*</td>
</tr>
</tbody>
</table>

What’s the updated running time?
End of Semester Logistics

**Lab:** Your final CS 225 lab is released today!
- No lab sections next week (partial week).

**Final Exam:** Final exams start on Reading Day (Dec 14)
- Last day of office hours is Wednesday, Dec. 13.
- No office/lab hours once the first final exam is given.

**Grades:** We’re working towards a “Pre-Final Update” including all current grades except your final exam.
- Expect this update early next week.
Shortest Path
Dijkstra's Algorithm (SSSP)

DijkstraSSSP(G, s):
6   foreach (Vertex v : G):
7       d[v] = +inf
8       p[v] = NULL
9       d[s] = 0
10
11      PriorityQueue Q // min distance, defined by d[v]
12      Q.buildHeap(G.vertices())
13      Graph T       // "labeled set"
14
15     repeat n times:
16        Vertex u = Q.removeMin()
17        T.add(u)
18     foreach (Vertex v : neighbors of u not in T):
19         if _______________ < d[v]:
20             d[v] = _______________
21             p[v] = m
Dijkstra’s Algorithm (SSSP)

What about negative weight cycles?
Dijkstra’s Algorithm (SSSP)

What about negative weight edges, without negative weight cycles?
Dijkstra’s Algorithm (SSSP)

What is the running time?

```plaintext
DijkstraSSSP(G, s):

6     foreach (Vertex v : G):
7         d[v] = +inf
8         p[v] = NULL
9         d[s] = 0
10
11     PriorityQueue Q // min distance, defined by d[v]
12     Q.buildHeap(G.vertices())
13     Graph T         // "labeled set"
14
15     repeat n times:
16         Vertex u = Q.removeMin()
17         T.add(u)
18     foreach (Vertex v : neighbors of u not in T):
19         if _______________ < d[v]:
20             d[v] = _______________
21             p[v] = m
```
Exam 12 (programming) is ongoing
More Info: https://courses.engr.illinois.edu/cs225/fa2017/exams/

MP7: The final MP!
Due: Monday, Dec. 11 at 11:59pm

Lab: lab_ml released today
lab_ml: Due Sunday @ 11:59pm

New POTDs every M/W/F
Worth +1 Extra Credit /problem (up to +40 total)