What is left

• Last Lab Starts today due Sunday
• Last Exam Friday in CBTF during lecture time
• Last MP finished yesterday
• 24-hour extensions
  • run for mp_intro, mp_stickers, mp_list*, mp_mosaics
  • run for mp_traversals and mp_mazes run by weekend
• 90% Regrade
  • form will be posted on Monday due by Wednesday.
  • Will grade the code in the repo on Wednesday May 5th at 11:59pm.
• Final Project Done by May 12th
  (This is a hard deadline due to timeline to grade)
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     PriorityQueue Q // min distance, defined by d[v]
11     Q.buildHeap(G.vertices())
12     Graph T        // "labeled set"
13
14     repeat n times:
15         Vertex u = Q.removeMin()
16         T.add(u)
17     foreach (Vertex v : neighbors of u not in T):
18         if cost(u, v) + d[u] < d[v]:
19             d[v] = cost(u, v) + d[u]
20             p[v] = u
Dijkstra’s Algorithm (SSSP)

Dijkstra gives us the shortest path from our path (single source) to **every** connected vertex!

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>p:</td>
<td>--</td>
<td>A</td>
<td>B</td>
<td>B</td>
<td>F</td>
<td>A</td>
<td>F</td>
<td>C</td>
</tr>
<tr>
<td>d:</td>
<td>0</td>
<td>10</td>
<td>17</td>
<td>15</td>
<td>12</td>
<td>7</td>
<td>11</td>
<td>21</td>
</tr>
</tbody>
</table>
**Dijkstra’s Algorithm (SSSP)**

**Q:** How does Dijkstra handle a single heavy-weight path vs. many light-weight paths?
Dijkstra’s Algorithm (SSSP)

**Q:** How does Dijkstra handle a single heavy-weight path vs. many light-weight paths?
Dijkstra’s Algorithm (SSSP)

Q: How does Dijkstra handle a single heavy-weight path vs. many light-weight paths?
Dijkstra’s Algorithm (SSSP)

Q: How does Dijkstra handle undirected graphs?
Dijkstra’s Algorithm (SSSP)

Q: How does Dijkstra handle negative weight cycles?

Shortest Path (A $\rightarrow$ E): $A \rightarrow F \rightarrow E \rightarrow (C \rightarrow H \rightarrow G \rightarrow E)^*$

Length: 12 \hspace{1cm} Length: -5 (repeatable)
Dijkstra’s Algorithm (SSSP)

Q: How does Dijkstra handle negative weight edges, without a negative weight cycle?
Dijkstra’s Algorithm (SSSP)

**Q:** How does Dijkstra handle negative weight edges, without a negative weight cycle?
Dijkstra's Algorithm (SSSP)

What is Dijkstra's running time?

```plaintext
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 u = Q.removeMin()
        T.add(u)
        foreach (Vertex v : neighbors of u not in T):
            if cost(u, v) + d[u] < d[v]:
                d[v] = cost(u, v) + d[u]
                p[v] = u

    return T
```
Landmark Path Problem

Suppose you want to travel from A to G.

Q1: What is the shortest path from A to G?
Landmark Path Problem

Suppose you want to travel from A to G.

Q2: What is the fastest algorithm to use to find the shortest path?
Landmark Path Problem

In your journey between A and G, you also want to visit the landmark L.

Q3: What is the shortest path from A to G that visits L?
Landmark Path Problem

In your journey between A and G, you also want to visit the landmark L.

Q4: What is the fastest algorithm to find this path?
Q5: What are the specific call(s) to this algorithm?