Paths in Graphs

BFS

What is length of path a → c? 1

DFS

What length of path from a to c? 3
Breadth First Search

def bfs(G, u):
    \( G = V, E \)
    \( u = \text{start node} \)

    for \( v \in V \):
        \( \text{dist}[v] = \infty \)
        \( \text{prev}[v] = \text{nil} \)
        \( \text{dist}[u] = 0 \)
        Q.push(u)

    while ! Q.empty():
        \( u = Q.pop() \)

        for \( u, v \in E \):
            if \( \text{dist}[v] > \text{dist}[u] + 1 \):
                \( \text{dist}[v] = \text{dist}[u] + 1 \)
                Q.push(v)
                \( \text{prev}[v] = u \)
Weighted Graph

What is cost best path

a - c: 3
a - q: 9 ≠ 7 the actual best path length!

We need something better.
Dijkstra's Shortest Path Algorithm for Weighted Graphs

```python
def dijkstra(G, u):
    for v in V:
        dist[v] = float('inf')
        prev[v] = nil
    dist[u] = 0
    Q = make_queue(V, dist)
    while not Q.empty():
        # O(1|V|pop)
        u = Q.pop()
        for (u, v, e) in E:
            if dist[v] > dist[u] + w(u, v):
                dist[v] = dist[u] + w(u, v)
                prev[v] = u
                Q.update(v, dist[v])
        # O(1|E|update)
```

$G = (V, E), w_{ij} \geq 0$