How can we solve the single source shortest path algorithm in a weighted graph where all weights ≥ 0?

What goes wrong with BFS?

weight 3 + 4 + 2 = 9
Use of queue in BFS was to process vertices in order of distance from source.

Reachable from 5:

\[ 0 \quad 1 \quad 2 \quad 2 \quad 3 \quad \cdots \quad 3 \cdots \]

Picked the reachable vertex in queue (not yet “placed in shortest path tree”) with shortest distance from source when removed from queue.
Dijkstra's Single Source Shortest Path (Vertex S)

\[ S.\text{tracker} = \text{pq}.\text{addTrack}(0, S) \]

For all \( v \in V - \{ S \} \)

\[ v.\text{tracker} = \text{pq}.\text{addTrack}(\infty, v) \]

For all \( v \in V \)

\[ v.\text{parent} = \text{null} \]

While \( \neg \text{pq}.\text{isEmpty}() \)

\[ \text{tag} = \text{pq}.\text{minTag}() \]

If \( \text{tag} = \infty \) return \# remaining vertices (in pq) are not reachable from S

\[ u = \text{pq}.\text{extractMin}() \]

\[ u.\text{dist} = \text{tag} \]

For each edge from \( u \)

\[ v = u.\text{dest} \]

If \( (v.\text{tracker}.\text{inCollection}() \&\& \)

\[ u.\text{dist} + e.\text{weight} < v.'s \text{current tag} ) \]

\[ v.\text{tracker}.\text{update}(u.\text{dist} + e.\text{weight}) \]

\[ v.\text{parent} = e \]