1. Recall: FullSearch - \( k \)-limited, \( \frac{1}{2} \)-approximation SSSP for weighted graphs
   (a) \( k \)-limited: \( s-t \) path has length at most the length of any \( s-t \) path of size \( k \).
   (b) \( k \)-limited, \( \frac{1}{2} \)-approximate: \( s-t \) path has length at most \( 1\frac{1}{2} \) times the length of any \( s-t \) path of size \( k \).
   (c) Work: \( O(m \log L) \), where \( L \) is the sum of all edge lengths
   (d) Depth: \( O\left(\frac{m \log L}{p} + k\right)\)

2. Today: Single-Source Shortest Paths (SSSP) for directed graph with nonnegative integral weights
   (a) Problem: FullSearch still depends on the edge weights
   (b) Idea: Shrink the graph iteratively until edge weights are manageable

3. Shrinking the edge weights in \( \hat{G} \)
   (a) For each edge \( uv \), reduce length by adding a cost for leaving a node:
      \[ \hat{l}(uv) = l(uv) + c(u) - c(v) \]
   (b) For a path \( P \) from source \( s \) to \( t \):
      \[ \hat{l}(P) = c(s) + l(P) - c(t) \]
   (c) Adding cost function preserves the shortest paths
   (d) When the cost of \( s \) is 0, \( \hat{l}(P) \leq l(P) \)
   (e) Assign \( c(v) \) based on \( d(v) \), but we don’t know exact \( d(v) \)
   (f) Idea: Use estimated distance instead

4. Estimating distances with \( \frac{1}{2} \)-approximate SSSP
   (a) \( \hat{l}(P) \) must stay nonnegative, so estimated distances must be underestimates
   (b) For any \( t \) distance \( d \) from \( s \), the estimated \( d(t) \) is such that \( \frac{1}{2}d \leq d(t) \leq d \)

5. Exact SSSP Algorithm: reduction to \( \frac{1}{2} \)-approximate SSSP
(a) Compute distance estimates $d(v)$ using $\frac{1}{2}$-approximate SSSP
(b) If all distance estimates are 0, return $d$
(c) Else, recurse on an auxiliary graph $\hat{G}$ with edge lengths:

$$\hat{l}(uv) = [d(u)] + l(uv) - [d(v)]$$

(d) Let $\hat{f}$ be the shortest path lengths in $\hat{G}$
(e) Return $f$:

$$f(v) = \hat{f}(v) + [d(v)]$$

6. Proof of correctness

(a) First, $\hat{l}$ remains nonnegative:

$$d(v) \leq d(u) + l(uv)$$

$$[d(v)] \leq [d(u)] + l(uv)$$

$$0 \leq \hat{l}(uv)$$

(b) For any path $P$ from source $s$ to node $t$:

$$\hat{l}(P) = \sum_{uv \in P} [d(u)] + l(uv) - [d(v)]$$

$$= [d(s)] + l(P) - [d(t)]$$

$$= l(P) - [d(t)]$$

(c) So $f(t)$ does return $l(P)$, where $P$ is the shortest path

7. Lemma: Recursion depth is log $L$

(a) Let $P$ be the shortest path from $s$ to $t$
(b) Maximum shortest path length in $\hat{G}$ is at most $\frac{L}{2}$:

$$\hat{l}(P) = l(P) - [d(x)]$$

$$\leq l(P) - \frac{1}{2}l(P) = \frac{1}{2}l(P)$$