Correctness of Dijkstra's algorithm:

Claim: for every vertex \( v \):

* at the moment when \( d(v) \) is minimum in \( Q \), we have
  \[ d(v) = \delta(s, v). \]
* from that moment, \( d(v) \) does not change any more.

1. At any moment: \( \delta(s,v) \leq d(v) \) for every vertex.
   
   **Proof**: as on page 93.

2. Assume at some moment, \( d(v) \) becomes equal to \( \delta(s,v) \). Then, during the rest of the algorithm, \( d(v) \) does not change.

   **Proof**: as on page 93.
Let \( v \neq s \). Consider the shortest path from \( s \) to \( v \).

Consider the iteration in which \( u \) is chosen as the vertex in \( Q \) whose \( d \)-value is minimum.

If \( d(u) = \delta(s, u) \) at the beginning of this iteration, then \( d(v) = \delta(s, v) \) at the end of this iteration.

**Proof:** As on page 94: At the end of this iteration,

\[
d(v) \leq d(u) + wt(u, v) \quad \text{// from algorithm}
= \delta(s, u) + wt(u, v) \quad \text{// from assumption}
= \delta(s, v) \quad \text{// property of shortest paths}
\leq d(v) \quad \text{// from 1}
\]

\[\therefore d(v) = \delta(s, v).\]
The minimum $d$-value in $Q$ never decreases.

**Proof:** Consider an iteration of the while-loop, and let $u \in Q$ be such that $d(u)$ is minimum.

Before the for-loop: $d(u) \leq d(v)$ for all $v \in Q$.

During the for-loop, some values $d(v)$ may change.

If $d(v)$ is changed, its new value is

$$d(v) = d(u) + wt(u,v) > d(u).$$

$\therefore$ At the end of the for-loop: $d(u) \leq d(v)$ for all $v \in Q$.

$\therefore$ In the next iteration of the while-loop:

all $d$-values in $Q \setminus \{u\}$ are $\geq d(u)$

\[\text{this is the new set } Q^{\text{new}}\]

$\therefore$ minimum $d$-value in $Q^{\text{new}}$ is $\geq d(u)$.

$\square$
Proof of the claim on page 102:
First observe: in each iteration of the while-loop, the set $Q$ gets smaller.

For each vertex $v$, at some moment, $d(v)$ is minimum over all vertices in $Q$.

For $v = s$: the claim is true.

Consider a vertex $v$ with $v \neq s$. Consider the shortest path $P$ from $s$ to $v$:

$$S \rightarrow u_1 \rightarrow u_2 \rightarrow u_3 \rightarrow \ldots \rightarrow u_k \rightarrow v$$

Observe: for each $i$ with $1 \leq i \leq k$: the path $S \rightarrow u_1 \rightarrow u_2 \rightarrow \ldots \rightarrow u_i$ is the shortest path from $s$ to $u_i$. 
Consider the first iteration of the while-loop:

Vertex $s$ is chosen.

At that moment: $d(s) = 0 = \delta(s,s)$.

From (3): at the end of the iteration in which $s$ is chosen as the vertex in $Q$ with minimum $d$-value: $d(u_1) = \delta(s,u_1)$.

From (2): $d(u_1)$ does not change afterwards.

Observe: $u_1$ is still in $Q$ at the end of the iteration in which $u_1$ is chosen: $d(u_1) = \delta(s,u_1)$.

From (3): At the end of the iteration in which $u_1$ is chosen: $d(u_2) = \delta(s,u_2)$.

From (2): $d(u_2)$ does not change afterwards.

Observe: $d(u_2) = \delta(s,u_2) = \delta(s,u_1) + \omega(t(u_1,u_2)) > \delta(s,u_1) = d(u_1)$

From (4): At the end of the iteration in which $u_1$ is chosen: $u_2$ is still in $Q$. 

At the beginning of the iteration in which \( u_2 \) is chosen: \( d(u_2) = \delta(s, u_2) \).

From (3): At the end of the iteration in which \( u_2 \) is chosen: \( d(u_3) = \delta(s, u_3) \).

From (2): \( d(u_3) \) does not change afterwards.

Observe: \( d(u_3) = \delta(s, u_3) = \delta(s, u_2) + \omega(t(u_2, u_3)) \geq \delta(s, u_2) = d(u_2) \)

From (4): At the end of the iteration in which \( u_2 \) is chosen: \( u_3 \) is still in \( Q \).

At the beginning of the iteration in which \( u_k \) is chosen: \( d(u_k) = \delta(s, u_k) \).

From (3): At the end of the iteration in which \( u_k \) is chosen: \( d(v) = \delta(s, v) \).

From (2): \( d(v) \) does not change afterwards.
Observe: \( d(v) = \delta(s,v) = \delta(s,u_k) + wt(u_k,v) \) 

\[ \geq \delta(s,u_k) = d(u_k) \]

From (4) : at the end of the iteration in which \( u_k \) is chosen: \( v \) is still in \( Q \).

\[ \therefore \text{the answer} \] At the beginning of the iteration in which \( v \) is chosen: \( d(v) = \delta(s,v) \). From (2) : \( d(v) \) does not change afterwards.