Lecture 16 - Nov 10

Graphs

Dijkstra's Algorithm: Tracing
Dijkstra's Algorithm: Pre- and Post-cond.

Announcements/Reminders

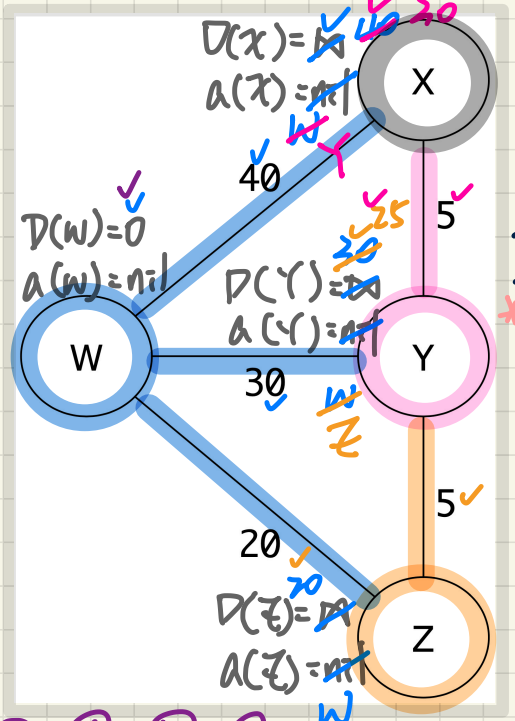
- Today's class: notes template posted
- Test 1 results released on Tuesday (Nov 4)
- Change of Dates:
  - + Assignment 2 to be released on Wed, Nov 12
  - + Assignment 2 to be due on Wed, Nov 19
  - + Test 2 to be take place on Mon, Nov 24

Dijkstra's Shortest Path Algorithm: Example 2

single source.

* # Iterations = $|V|$

** remove X from Q



$d(W,W) = 0$
$d(W,X) = 30$
$d(W,Y) = 25$
$d(W,Z) = 20$

Init.	W (src)	X	Y	Z	D(X) = d(W,X)
D	0	∞	∞	∞	
a	nil	nil	nil	nil	

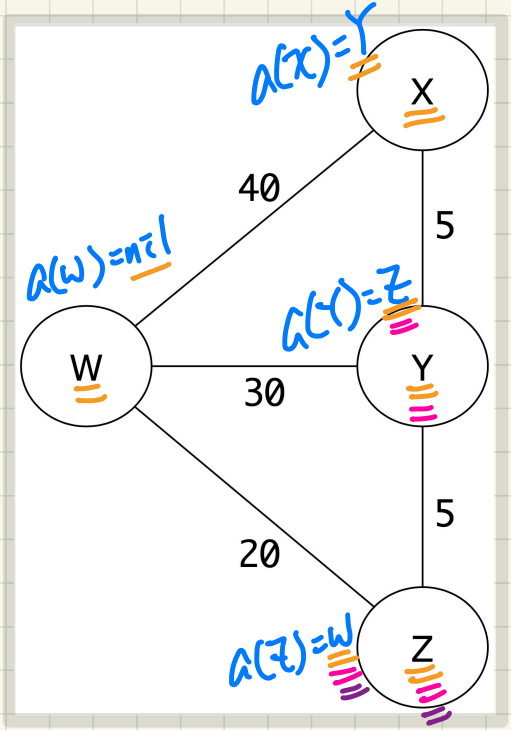
# Iteration	min D	changes
1	W D(W)=0	X, Y, Z in Q a(X)=W ← 0 + 40 < ∞ → D(X) = 40 ≠ d(W,X) a(Y)=W ← 0 + 30 < ∞ → D(Y) = 30 ≠ d(W,Y) a(Z)=W ← 0 + 20 < ∞ → D(Z) = 20 = d(W,Z) (not final) remove W from Q. → D(W) = d(W,W) final
2	Z D(Z)=20	W ∉ Q → not considered. a(Y)=Z ← 20 + 5 < 30 → D(Y) = 25 = d(W,Y) not final remove Z from Q → D(Z) = d(W,Z) final
3	Y D(Y)=25	W ∉ Q → not considered. a(X)=Y ← 25 + 5 < 40 → D(X) = 30 = d(W,X) not final remove Y from Q → D(Y) = d(W,Y) final
4	X D(X)=30	W ∉ Q ∧ Y ∉ Q → not considered. **

Q*

①	②	③	④
W	Z	Y	X

* min key (D value) priority queue

Upon termination of Dijkstra's algorithm

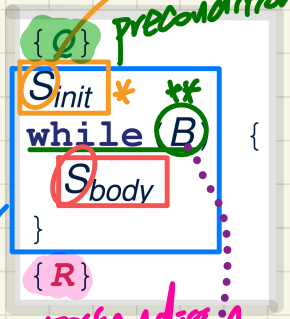


dest.	ancestor path.
x	x y z w
y	y z w
z	z w

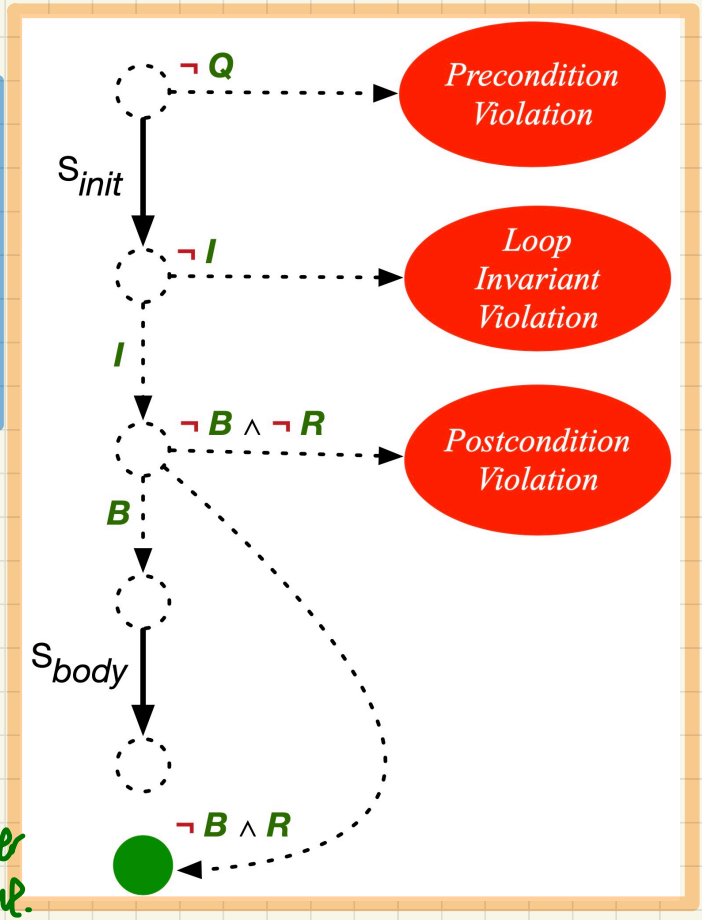
dest.	shortest path (from source w)
x	w z y x
y	w z y
z	w z

Correctness of Loops: Syntax

programming statements.




```
void myAlgorithm() {
  assert Q; /* Precondition */
  Sinit
  assert I; /* Is LI established? */
  while( B ) {
    Sbody
    assert I; /* Is LI preserved? */
  }
  assert R; /* Postcondition */
}
```



Iterative Algorithm

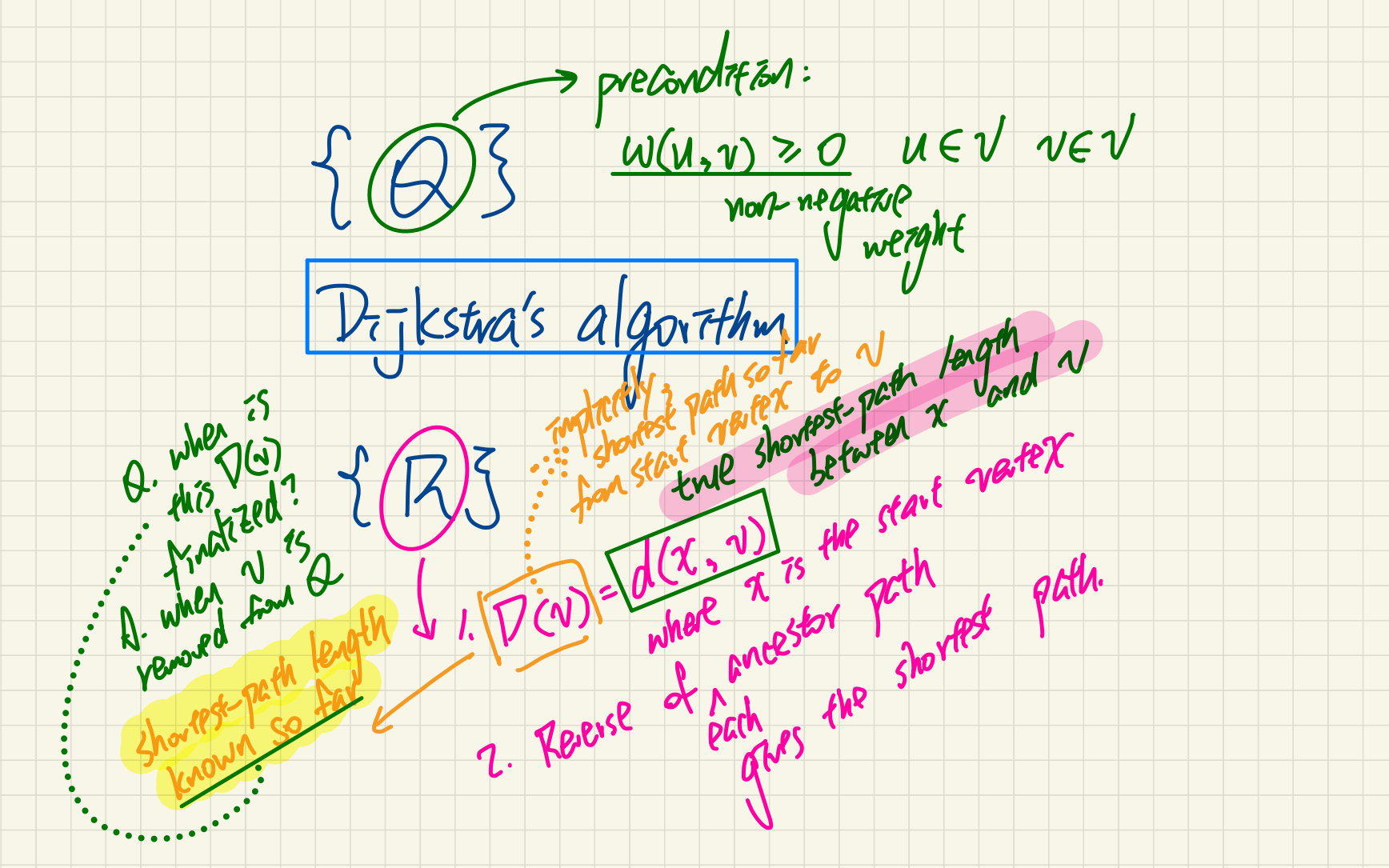
B: stay condition

¬B: exit condition.

* Initialization/Preparation for iterations.

** As long as B is true, execute Sbody another time.

As soon as B is false, exit from the loop.


precondition:
$w(u,v) \geq 0 \quad u \in V \quad v \in V$
non-negative weight
{Q}
Dijkstra's algorithm
{R}
implicitly, shortest path so far from start vertex to v
true shortest-path length between x and v
Q. when is this D(v) finalized?
A. when v is removed from Q
shortest-path length known so far
1. $D(v) = d(x,v)$
where x is the start vertex
2. Reverse of ancestor path each gives the shortest path.