CSE 105:
Data Structures and Algorithms-I
(Part 2)

Instructor

Dr Md Monirul Islam

Graph Searching

Breadth-First Search

Review

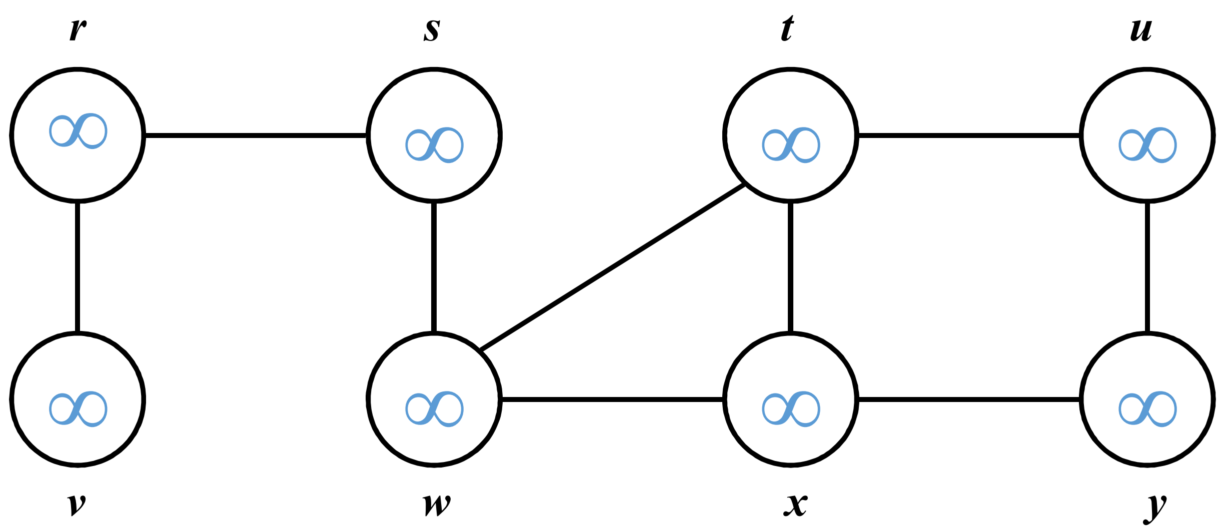
```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Whitening

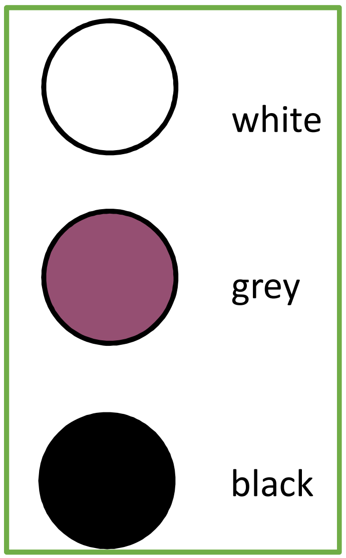
Enqueue the root

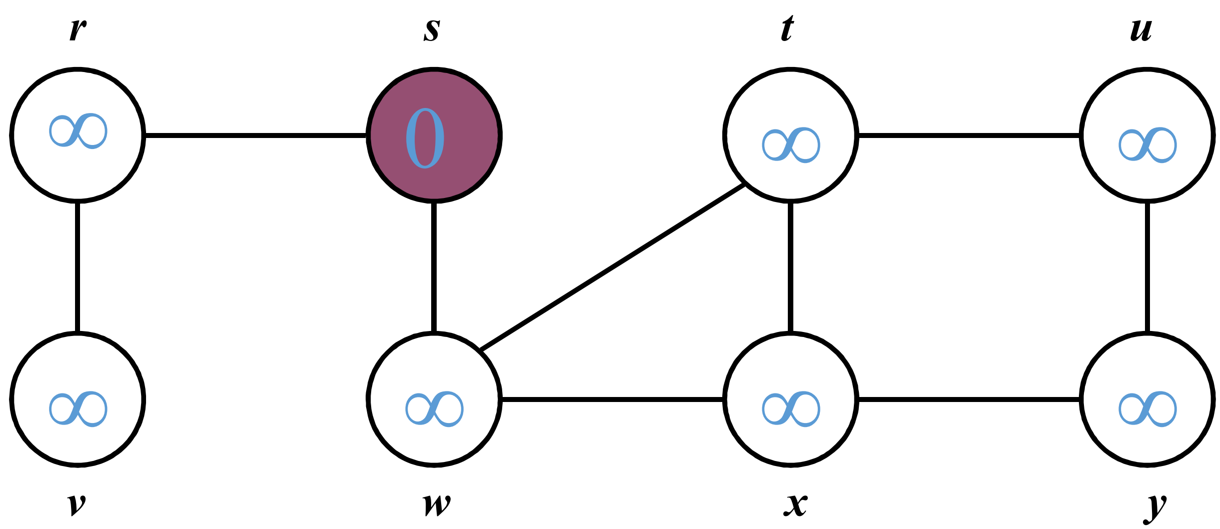
runs until queue is empty

Breadth-First Search: Example



Breadth-First Search: Example

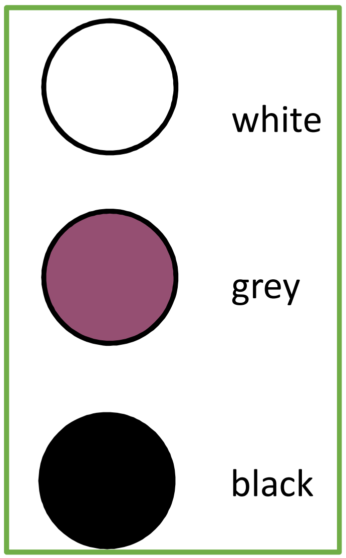


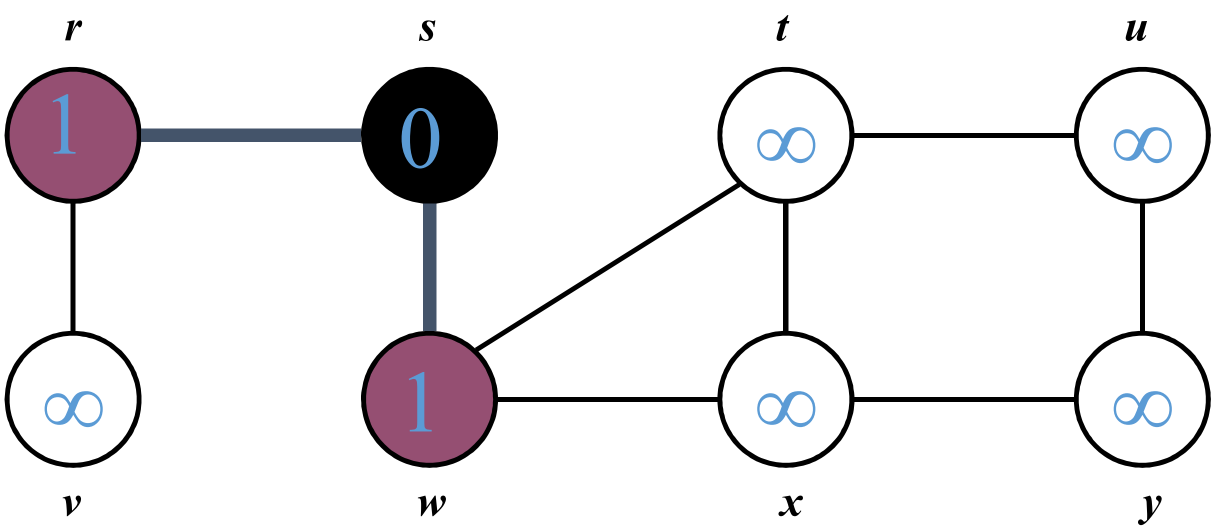


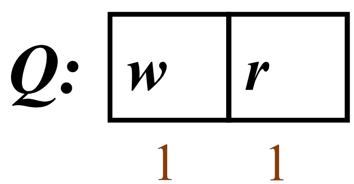
Q: | *s* |
0

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Breadth-First Search: Example

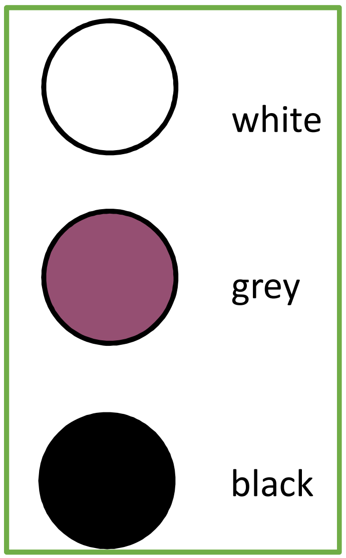


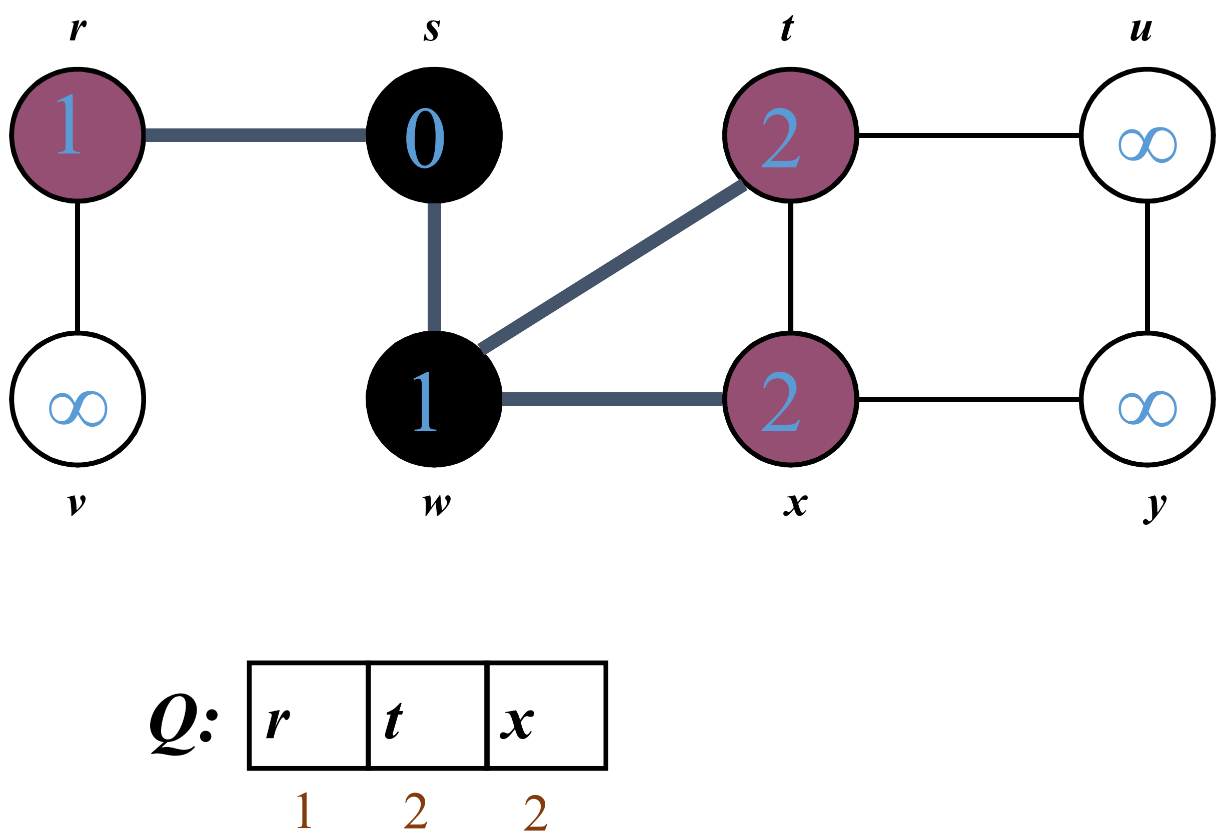





```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

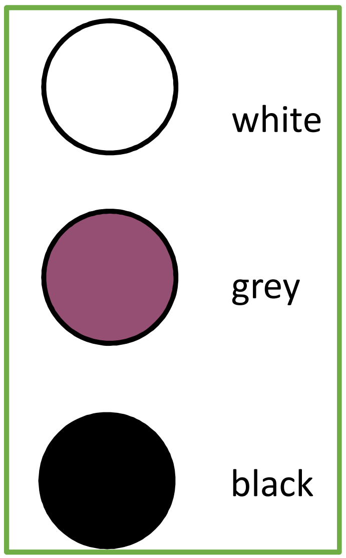
Breadth-First Search: Example

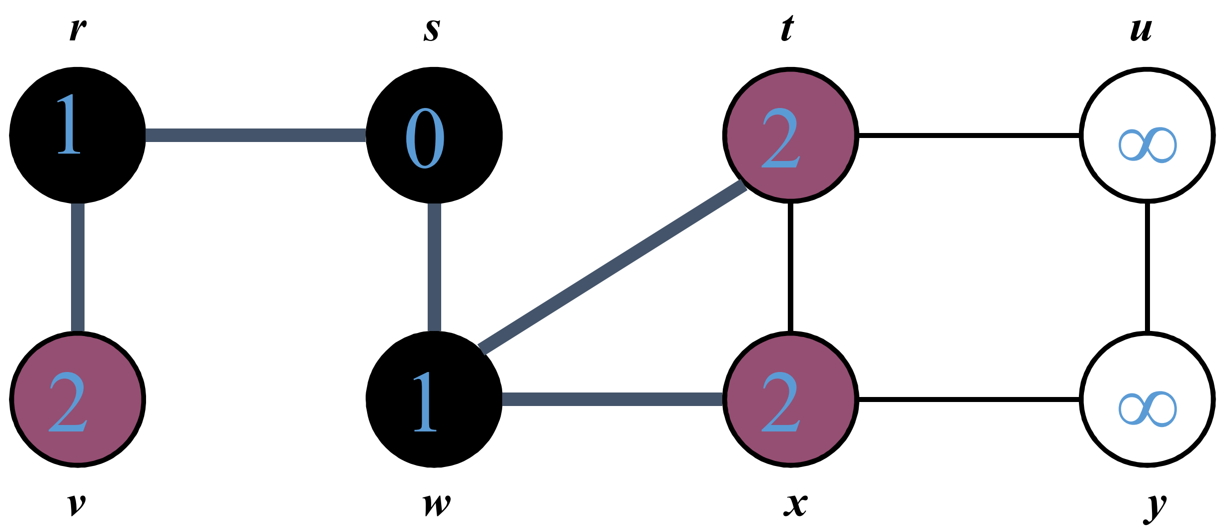


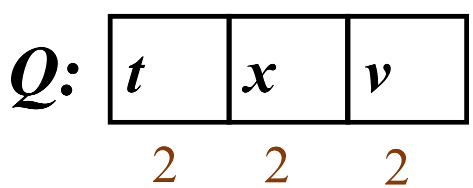



```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Breadth-First Search: Example

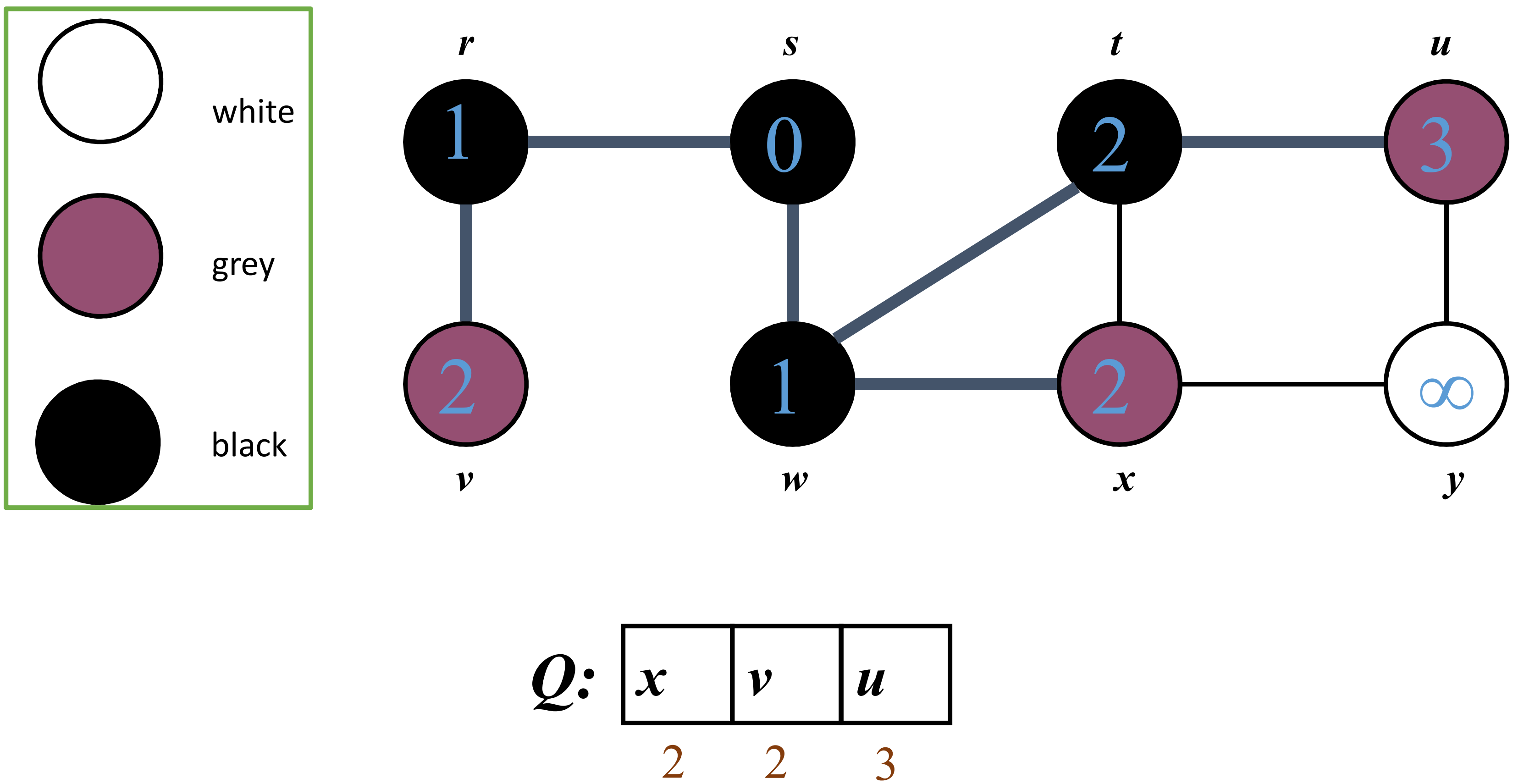






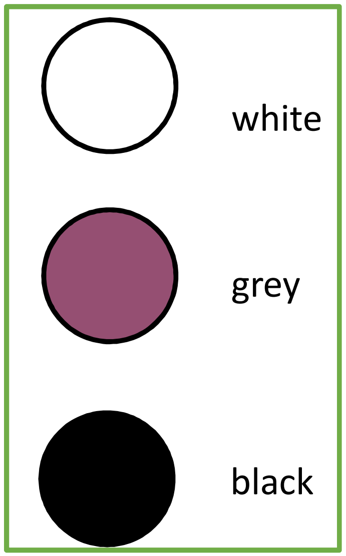

```
BFS(G, s)
  for each vertex u ∈ G.V − {s}
      u.color = WHITE
      u.d = ∞
      u.π = NIL
  s.color = GRAY
  s.d = 0
  s.π = NIL
  Q = ∅
  ENQUEUE(Q, s)
  while Q ≠ ∅
      u = DEQUEUE(Q)
      for each v ∈ G.Adj[u]
          if v.color == WHITE
              v.color = GRAY
              v.d = u.d + 1
              v.π = u
              ENQUEUE(Q, v)
      u.color = BLACK
```

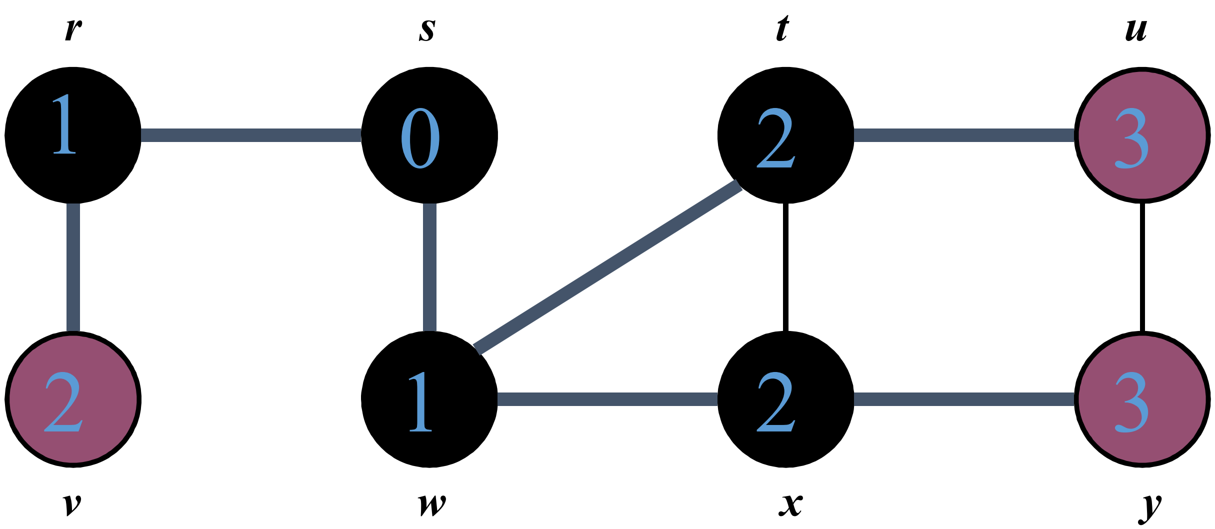
Breadth-First Search: Example




```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Breadth-First Search: Example



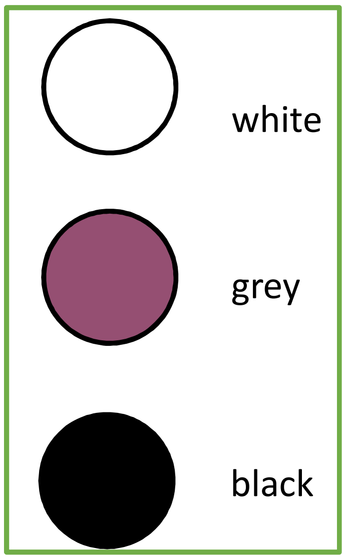


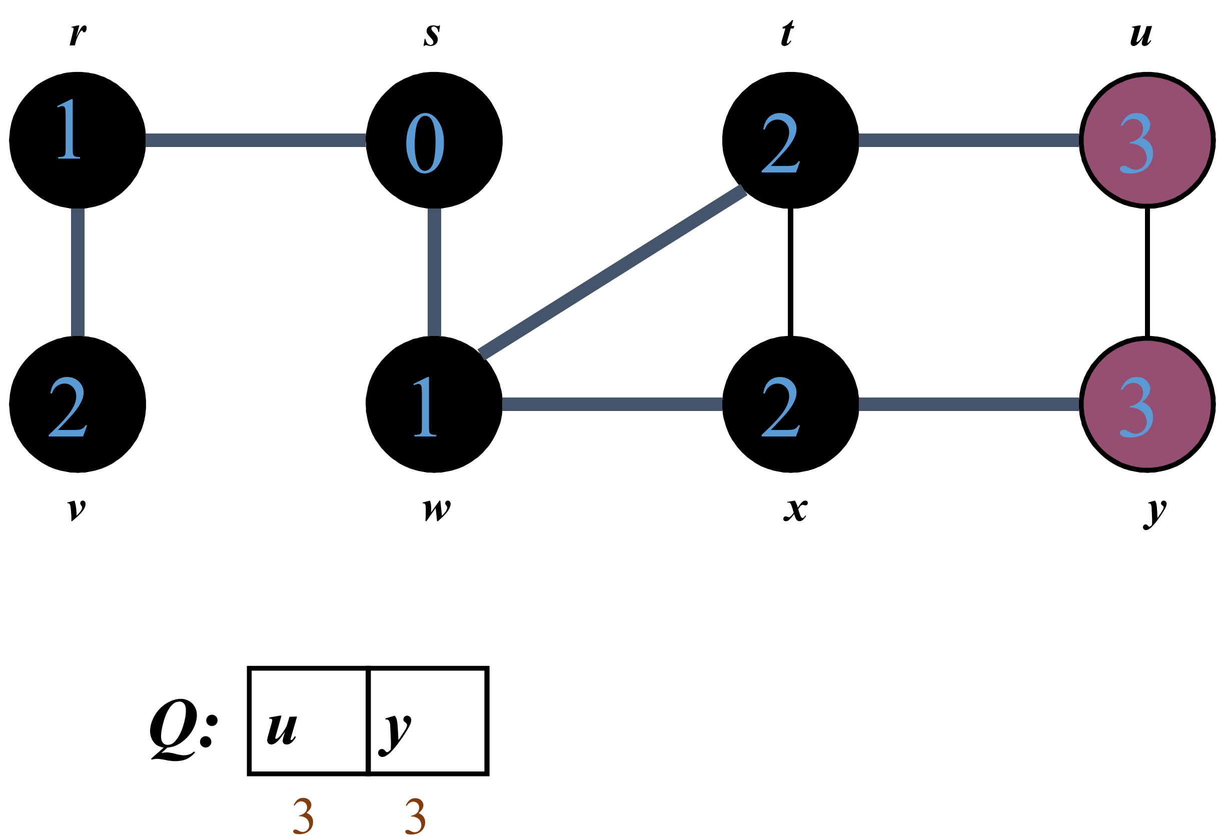
Q:

v	*u*	*y*
2	3	3

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

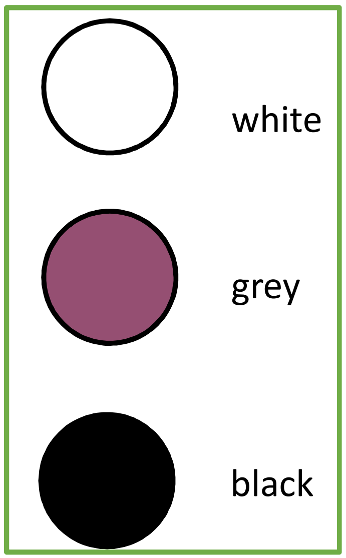
Breadth-First Search: Example

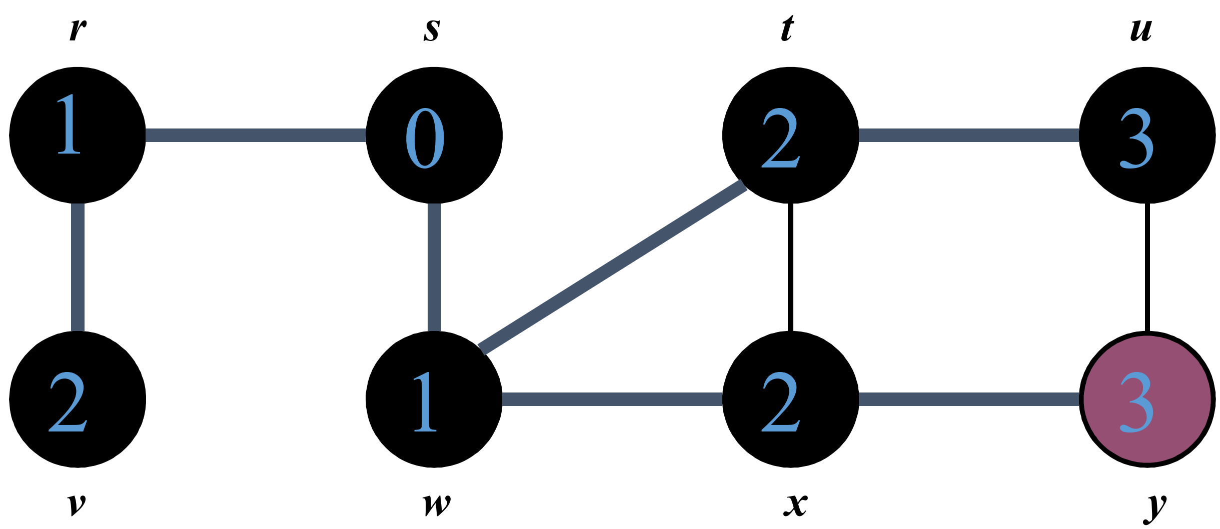





```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Breadth-First Search: Example



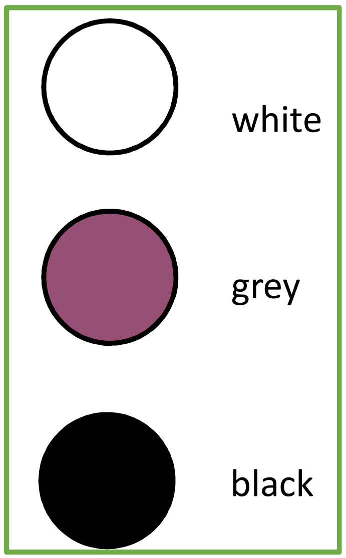


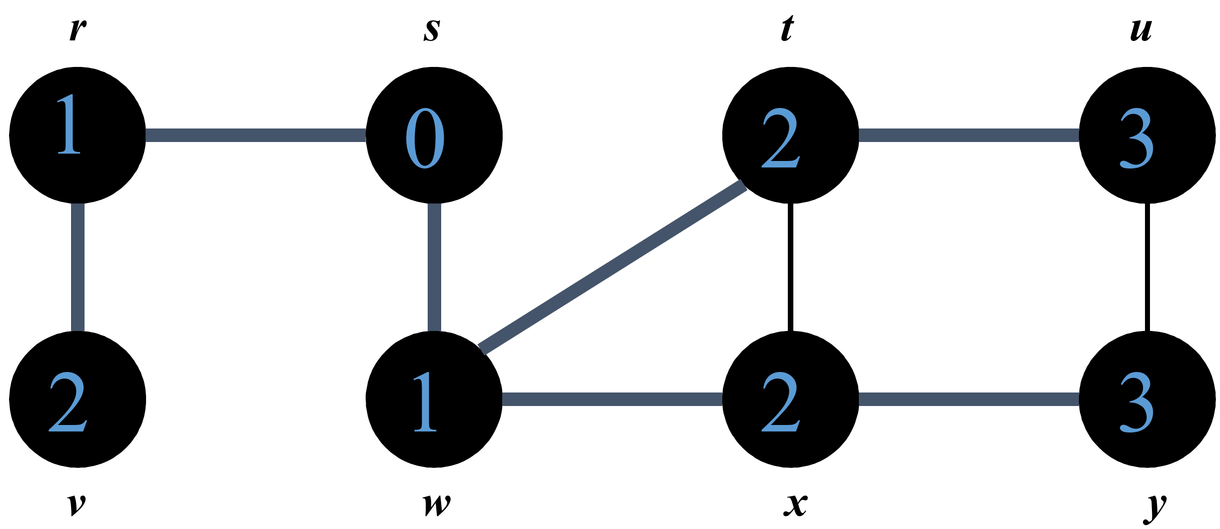
Q: $\boxed{y}$

3

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = Ø
ENQUEUE(Q, s)
while Q ≠ Ø
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Breadth-First Search: Example





***Q*: Ø**

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = Ø
ENQUEUE(Q, s)
while Q ≠ Ø
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

BFS: Analysis

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

$O(V)$

$O(1)$

BFS: Analysis

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

O(V)

O(1)

Each vertex is enqueued/dequeued once: O(V) in total

Once dequeued, the adjacency list of a vertex is explored: O(E) in total

BFS: Analysis

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

O(V)

O(1)

Each vertex is enqueued/dequeued once: O(V) in total

Once dequeued, the adjacency list of a vertex is explored: O(E) in total

O(V+E)

Breadth-First Search: Properties

- BFS calculates the *shortest-path distance* from the source node
  - *Shortest-path distance* $\delta(s, v)$

    = minimum number of edges from s to v, OR

    = ∞, if v is NOT reachable from s

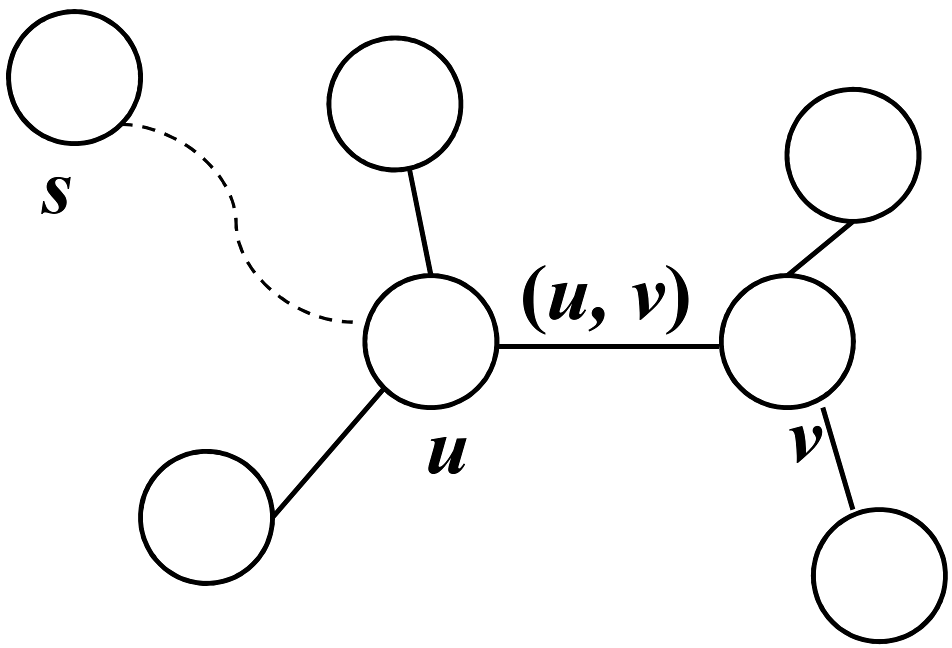
Breadth-First Search: Properties

- BFS calculates the *shortest-path distance* from the source node
  - *Shortest-path distance* $\delta(s, v)$
    = minimum number of edges from s to v, OR
    = ∞, if v is NOT reachable from s
- BFS builds *breadth-first tree*, in which paths from root represent shortest paths in G
  - Thus can use BFS to calculate shortest path from one vertex to another in $O(V+E)$ time in an unweighted graph

Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

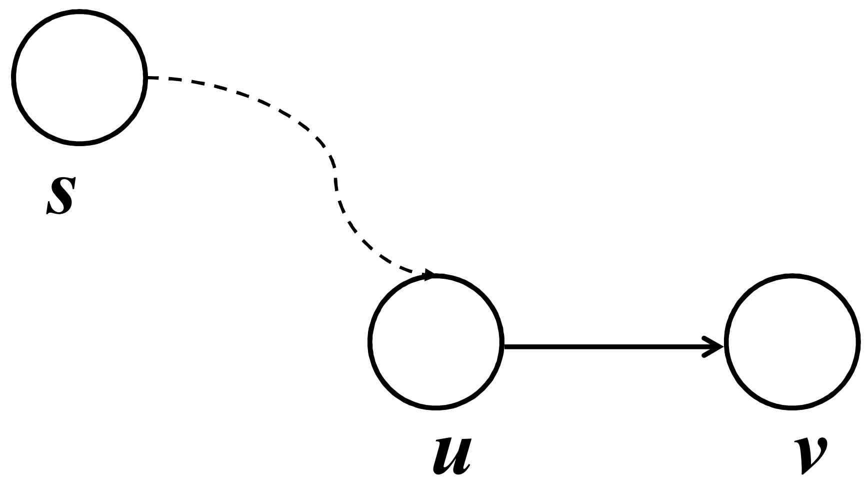


Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

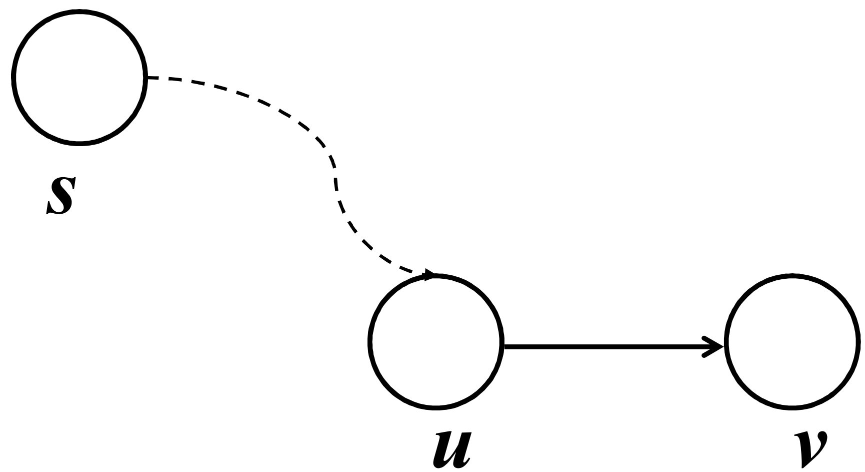
If u is reachable from s, so is v



Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

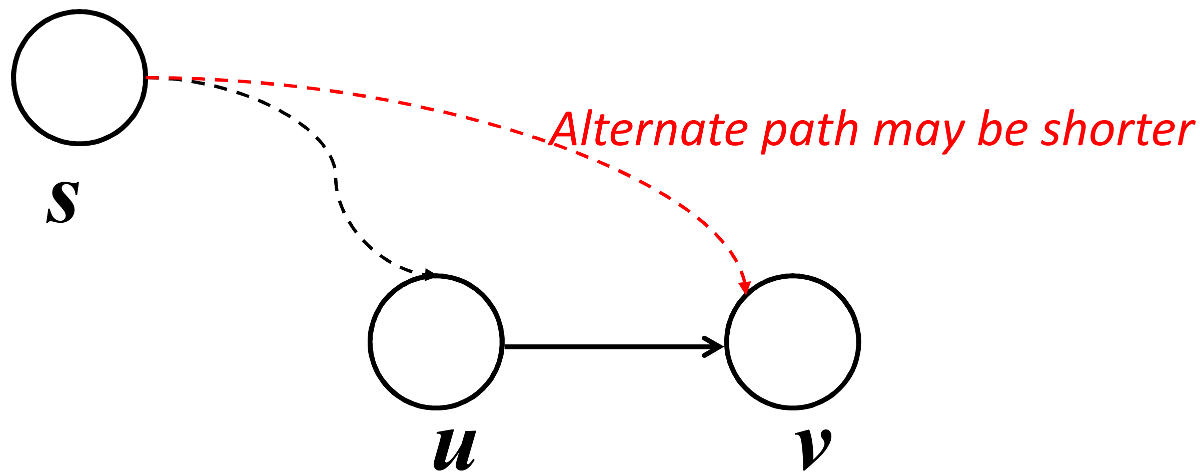


Shortest-path to v cannot be longer *than shortest path to u* plus edge (*u*, *v*)

Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$



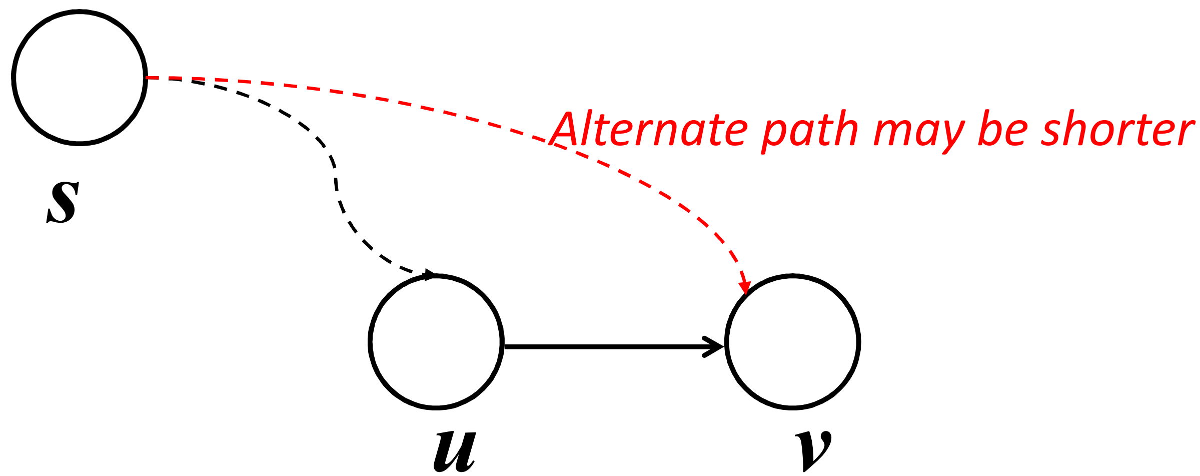
Shortest-path to v cannot be longer *than shortest path to* u plus edge (u, v)

Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let $s \in V$ be an arbitrary vertex. Then, for any edge $(u, v) \in E$,

$\delta(s, v) \leq \delta(s, u) + 1$

So we proved



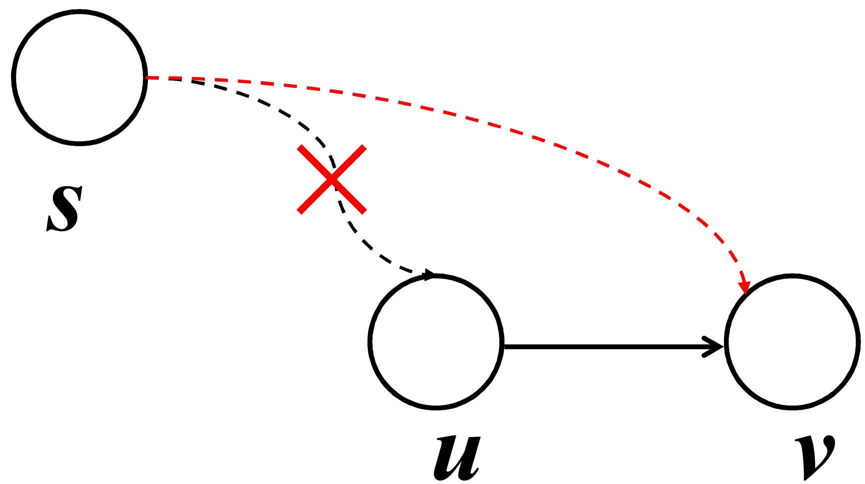
Shortest-path to v cannot be longer *than shortest path to* u plus edge (u, v)

Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

If u is NOT reachable from s

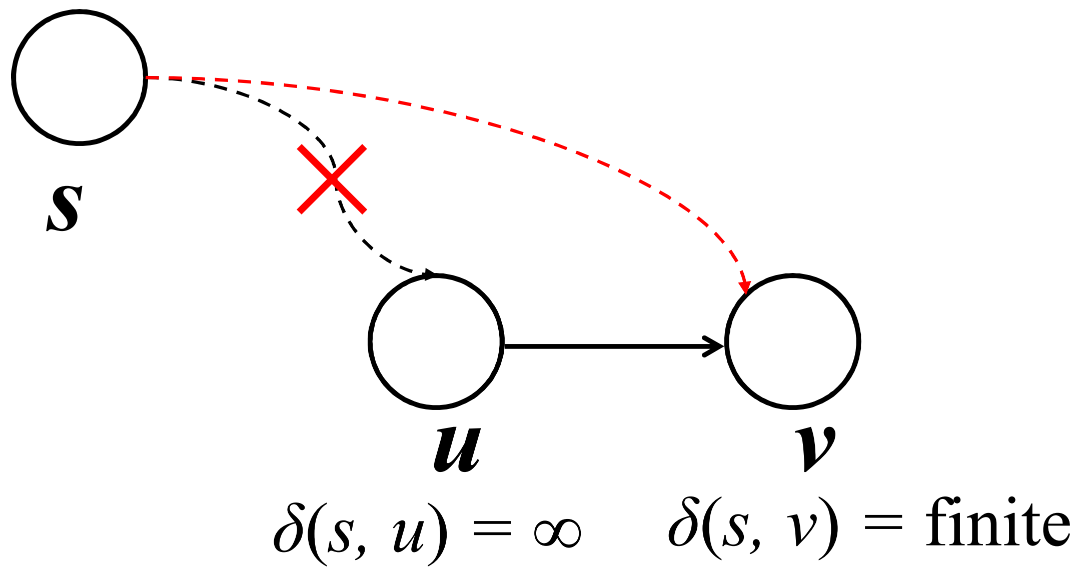


Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

If u is NOT reachable from s



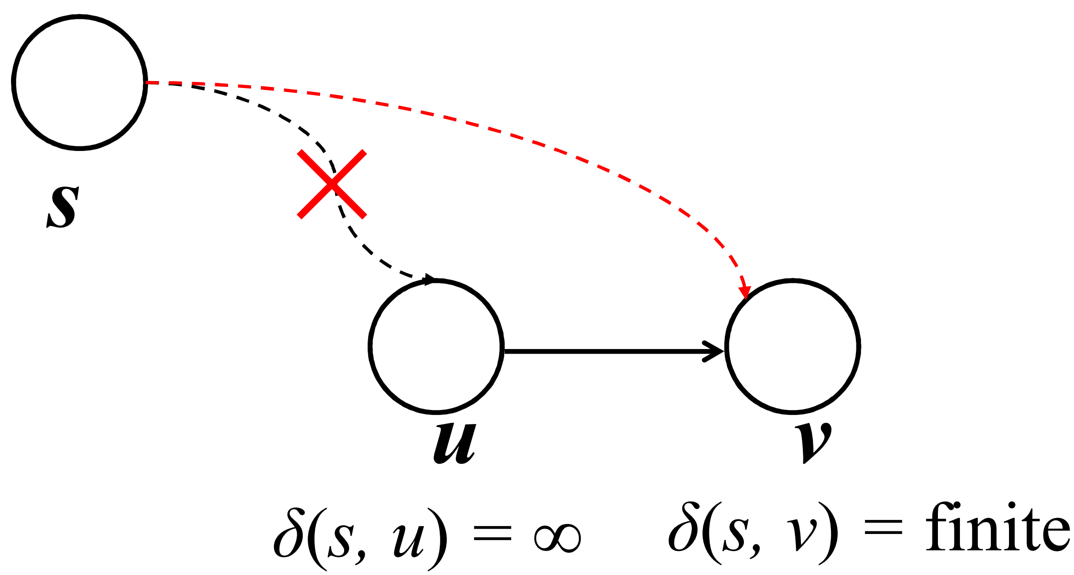
Lemma 22.1

Let G = (V, E) be a directed or undirected graph, and let s Ɛ V be an arbitrary vertex. Then, for any edge (u, v) Ɛ E,

$\delta(s, v) \leq \delta(s, u) + 1$

we proved again

If u is NOT reachable from s



```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
 Next       v.π = u
 EnQs       ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.2

Let $G = (V, E)$ be a directed or undirected graph, and suppose that BFS is run on G from a given source vertex $s \in V$. Then upon termination, for each vertex $v \in V$, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Next EnQs

Lemma 22.2

Let $G = (V, E)$ be a directed or undirected graph, and suppose that BFS is run on G from a given source vertex $s \in V$. Then upon termination, for each vertex $v \in V$, the value $v.d$ computed by BFS satisfies P

$v.d \geq \delta(s, v)$

Assume, before an EnQ, P holds

Then show, after the next EnQ, P still holds

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Next EnQs

Assume, before an EnQ, P holds

Then show, after the next EnQ, P still holds

Lemma 22.2

Then upon termination, for each vertex $v \in V$, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$ P

Basis:

$s.d = 0 = \delta(s, s)$ $\Rightarrow$ $s.d \geq \delta(s, s)$

and

$v.d = \infty \geq \delta(s, v)$ for all other vertices v

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
 Next       v.π = u
 EnQs       ENQUEUE(Q, v)
    u.color = BLACK
```

Assume, before an EnQ, P holds

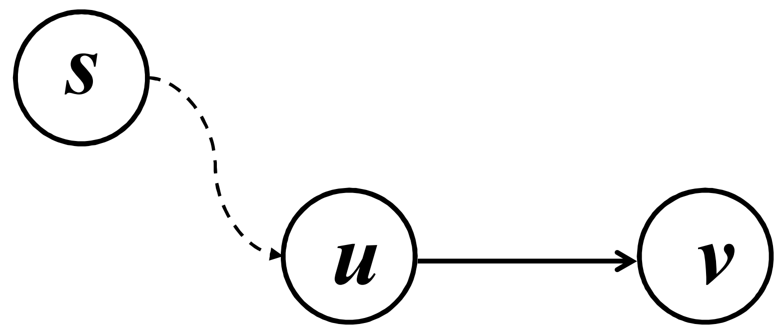
Then show, after the next EnQ, P still holds

Lemma 22.2

Then upon termination, for each vertex v Ɛ V, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$ P

Induction:

Let, white vertex v is discovered from u



```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Next EnQs

Assume, before an EnQ, P holds

Then show, after the next EnQ, P still holds

Lemma 22.2

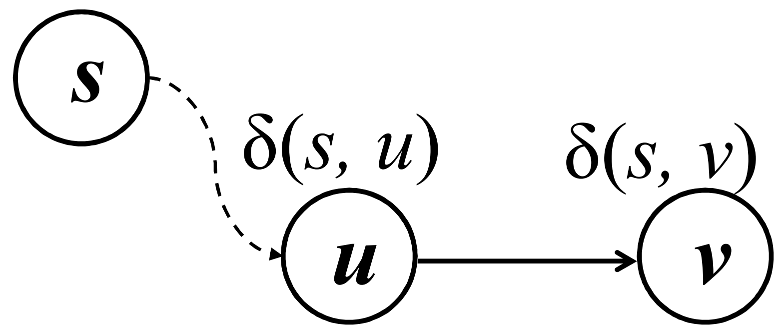
Then upon termination, for each vertex $v \in V$, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$ P

Induction:

Let, white vertex v is discovered from u

Now, by induction: $u.d \geq \delta(s, u)$

By lemma 22.1: $\delta(s, v) \leq \delta(s, u) + 1$



```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Next EnQs

Assume, before an EnQ, P holds

Then show, after the next EnQ, P still holds

Lemma 22.2

Then upon termination, for each vertex v Ɛ V, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$ P

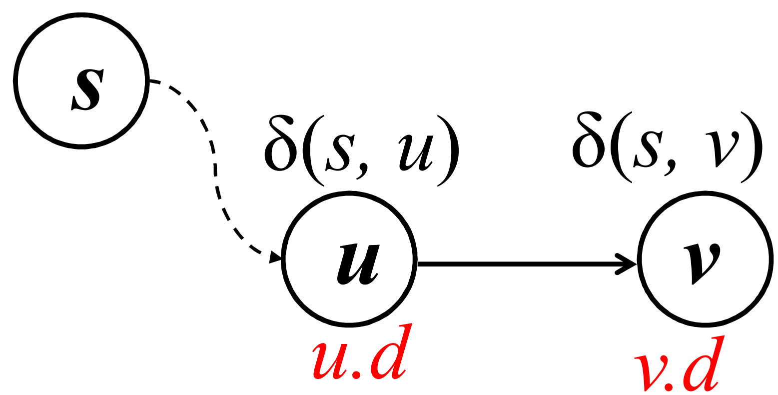
Induction:

Let, white vertex v is discovered from u

Now, by induction: $u.d \geq \delta(s, u)$

By lemma 22.1: $\delta(s, v) \leq \delta(s, u) + 1$

Now, $v.d = u.d + 1$



```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Next EnQs

Assume, before an EnQ, P holds

Then show, after the next EnQ, P still holds

Lemma 22.2

Then upon termination, for each vertex v Ɛ V, the value $v.d$ computed by BFS satisfies $v.d \geq \delta(s, v)$ P

Induction:

Let, white vertex v is discovered from u

Now, by induction: $u.d \geq \delta(s, u)$

By lemma 22.1: $\delta(s, v) \leq \delta(s, u) + 1$

$$\begin{aligned} \text{Now, } v.d &= u.d + 1 \\ &\geq \delta(s, u) + 1 \\ &\geq \delta(s, v) \end{aligned}$$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Suppose that during the execution of BFS on a graph $G = (V, E)$, the queue Q contains the vertices $<v_1, v_2, \cdots, v_r>$ where v_1 is the head of Q and v_r is the tail. Then (1) $v_r.d \le v_1.d + 1$ and (2) $v_i.d \le v_{i+1}.d$ for $i = 1, 2, \cdots, r-1$

v_1	v_2	v_3	.	..	.	.	.		v_r

Vertices in Queue

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

during the execution, the queue $Q = <v_1, v_2, \cdots, v_r>$

where v_1 = head and v_r = tail

Then (1) $v_r.d \leq v_1.d + 1$ and

(2) $v_i.d \leq v_{i+1}.d$ for $i = 1, 2, \cdots, r\text{-}1$

v_1	v_2	v_3	.	..	.	.	.		v_r

Vertices in Queue

We have to prove:

2. $v_1.d \leq v_2.d \leq v_3.d \ldots \leq v_r.d$

1. $v_r.d \leq v_1.d + 1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

$Q = <v_1, v_2, \cdots, v_r>$ where v_1 = head and v_r = tail

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_i.d \le v_{i+1}.d$ for $i = 1, 2, \cdots, r\text{-}1$

v_1	v_2	v_3	.	..	.	.	.		v_r

Basis:

It is true., as queue contains only s.

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

$Q = \langle v_1, v_2, \cdots, v_r \rangle$ where v_1 = head and v_r = tail

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_i.d \leq v_{i+1}.d$ for $i = 1, 2, \cdots, r-1$

v_1	v_2	v_3	.	..	.	.	.		v_r

We will prove both for Dequeue and Enqueue operations

1. $v_1.d \leq v_2.d \leq v_3.d \ldots \leq v_r.d$

2. $v_r.d \leq v_1.d + 1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r

Before DEQUEUE

Induction

Before DEQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r

Before DEQUEUE

v_2	v_3	.	..	.	.	.		v_r

After DEQUEUE

Induction

Before DEQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

After DEQUEUE (of v1):

$v_2.d <= v_3.d \ldots <= v_r.d$ (Okay)

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r	Before DEQUEUE

v_2	v_3	.	..	.	.	.		v_r	After DEQUEUE

Induction

Before DEQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

After DEQUEUE (of v1):

$v_2.d <= v_3.d \ldots <= v_r.d$ (Okay)

$v_r.d <= v_1.d + 1$ (from previous relation)

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r	Before DEQUEUE

v_2	v_3	.	..	.	.	.		v_r	After DEQUEUE

Induction

Before DEQUEUE, IH holds:

$\boxed{v_1.d <= v_2.d} <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

After DEQUEUE (of v1):

$v_2.d <= v_3.d \ldots <= v_r.d$ (Okay)

$v_r.d <= \boxed{v_1.d + 1 <= v_2.d + 1}$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r	Before DEQUEUE

v_2	v_3	.	..	.	.	.		v_r	After DEQUEUE

Induction

Before DEQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

After DEQUEUE (of v1):

$v_2.d <= v_3.d \ldots <= v_r.d$ (Okay)

$v_r.d <= v_1.d + 1 <= v_2.d + 1$ (Okay)

BFS(G, s)

```
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

v_1	v_2	v_3	.	..	.	.	.		v_r

Before ENQUEUE

Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

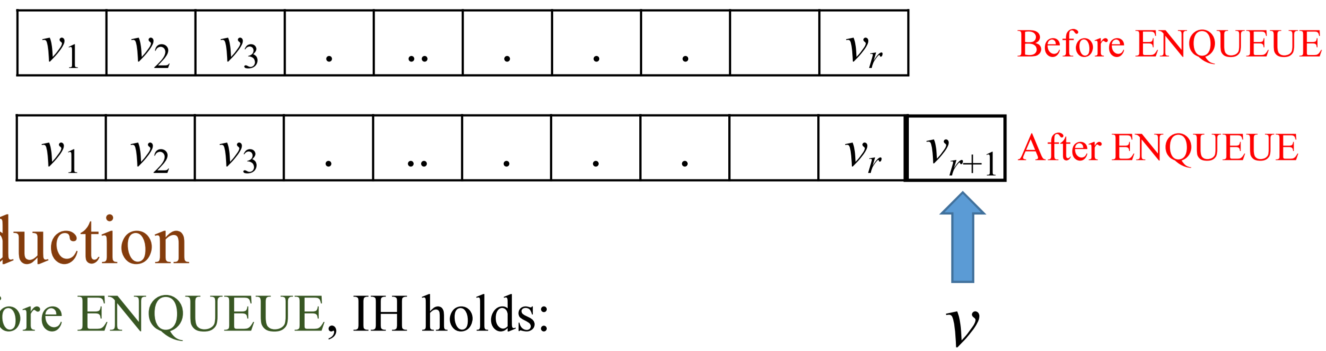
$v_r.d <= v_1.d + 1$

```
BFS(G, s)
    for each vertex u ∈ G.V − {s}
        u.color = WHITE
        u.d = ∞
        u.π = NIL
    s.color = GRAY
    s.d = 0
    s.π = NIL
    Q = Ø
    ENQUEUE(Q, s)
    while Q ≠ Ø
        u = DEQUEUE(Q)
        for each v ∈ G.Adj[u]
            if v.color == WHITE
                v.color = GRAY
                v.d = u.d + 1
                v.π = u
                ENQUEUE(Q, v)
        u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$



Induction

Before ENQUEUE, IH holds:

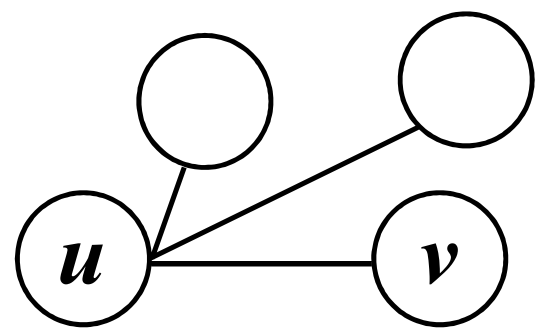
$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$

After ENQUEUE (of v):

Let, we enqueue v from u

v becomes v_{r+1}.

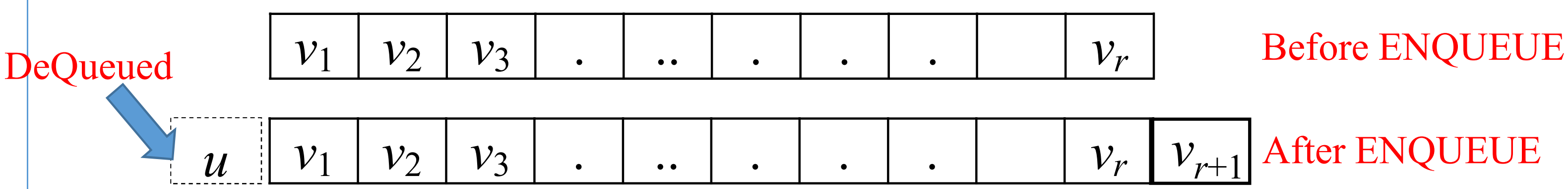


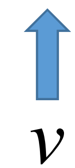
```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$



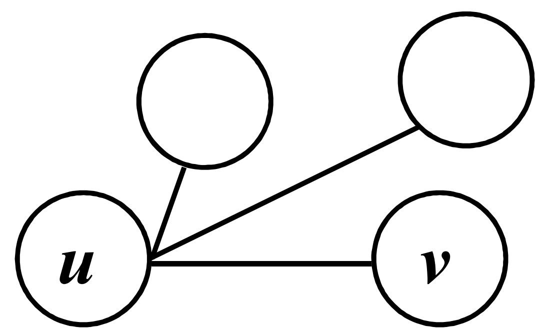


Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

u was is IN queue but dequeued

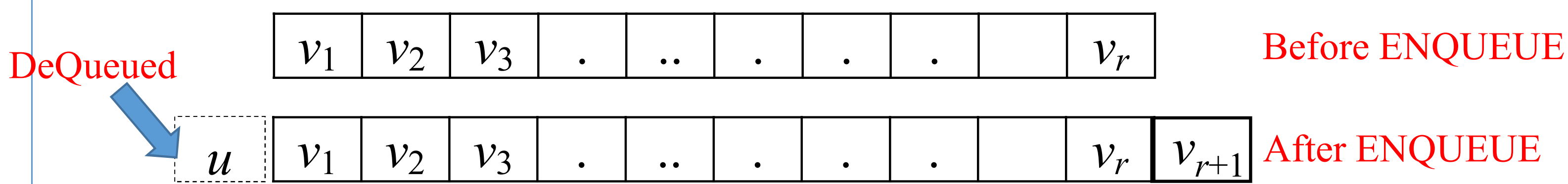
$u.d \leq v_1.d$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

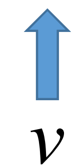
Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$



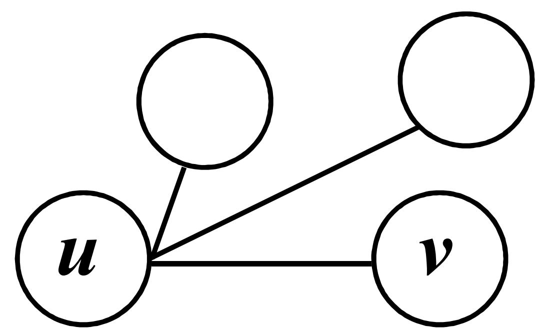
Induction



Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

u was is IN queue but dequeued

$u.d \le v_1.d$

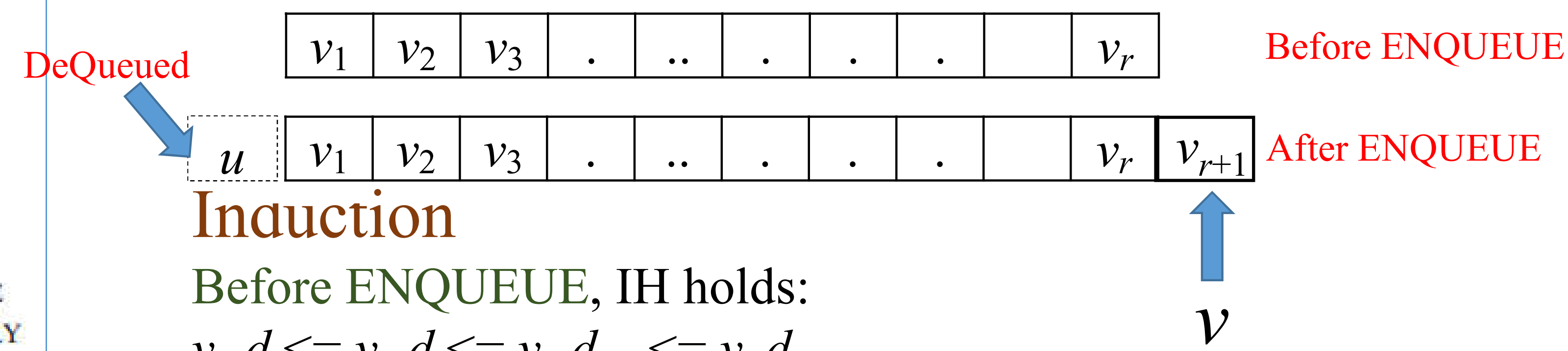
$v_{r+1}.d = v.d = u.d + 1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

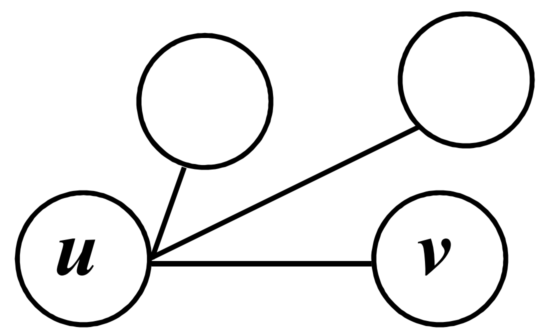


Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

u was is IN queue but dequeued

$u.d \leq v_1.d$

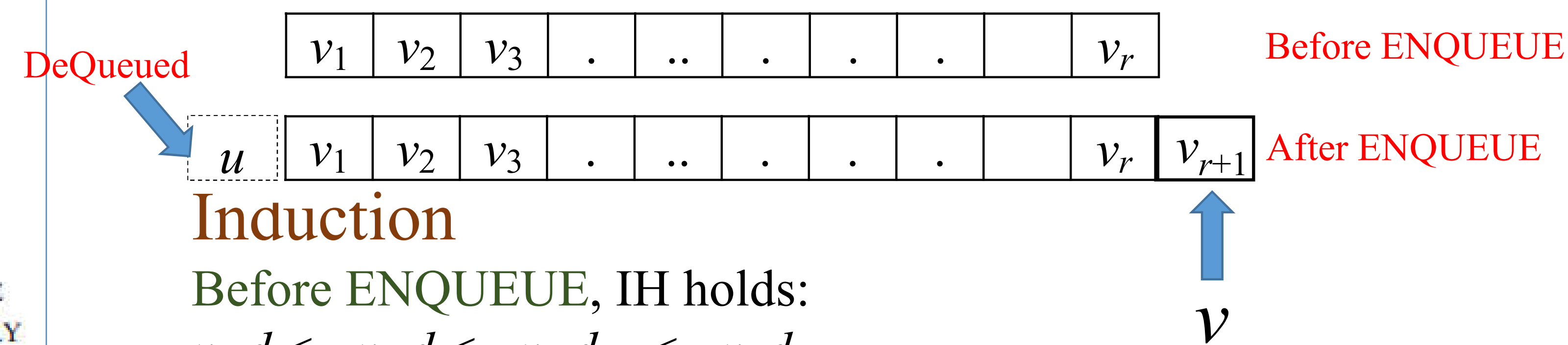
$v_{r+1}.d = v.d = u.d + 1 \leq v_1.d + 1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

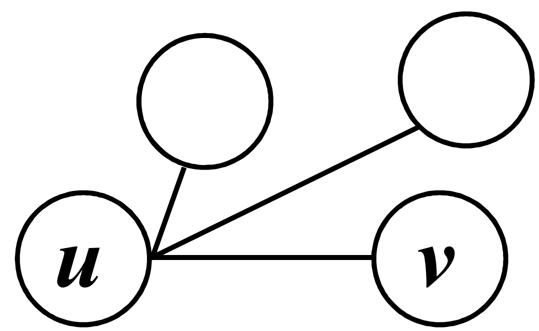


Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

u was is IN queue but dequeued

$u.d \leq v_1.d$

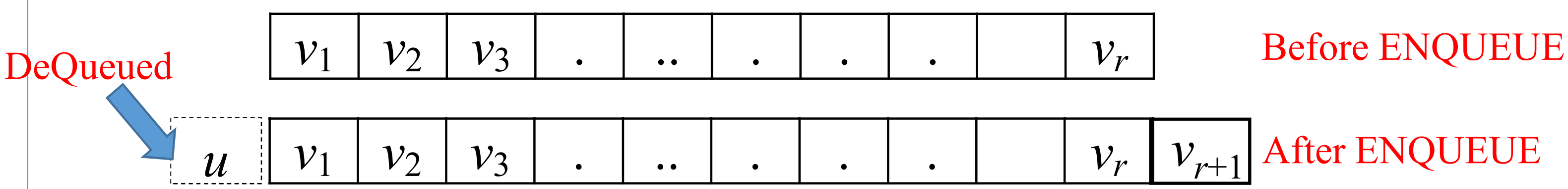
$v_{r+1}.d \leq v_1.d+1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

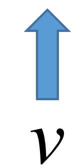
Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$



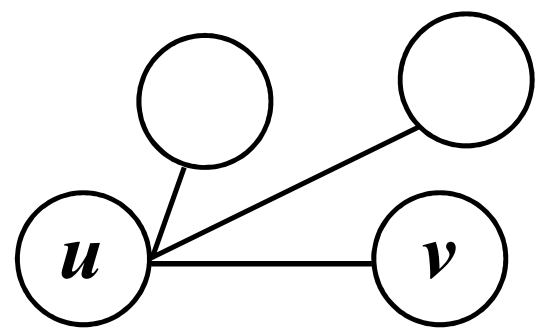
Induction



Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

By induction

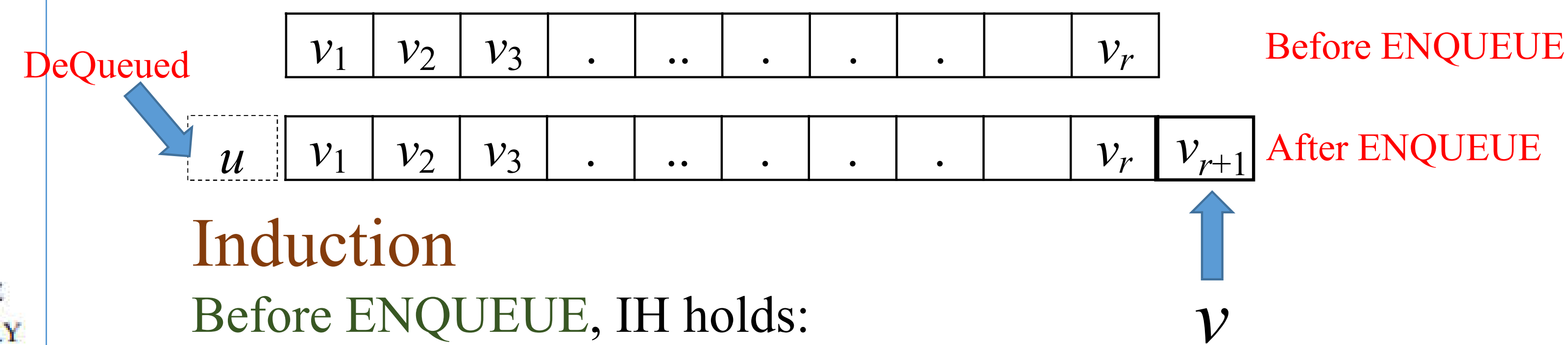
$v_r.d \le u.d{+}1$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

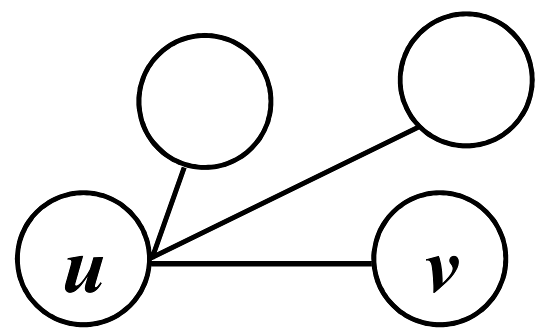


Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

By induction

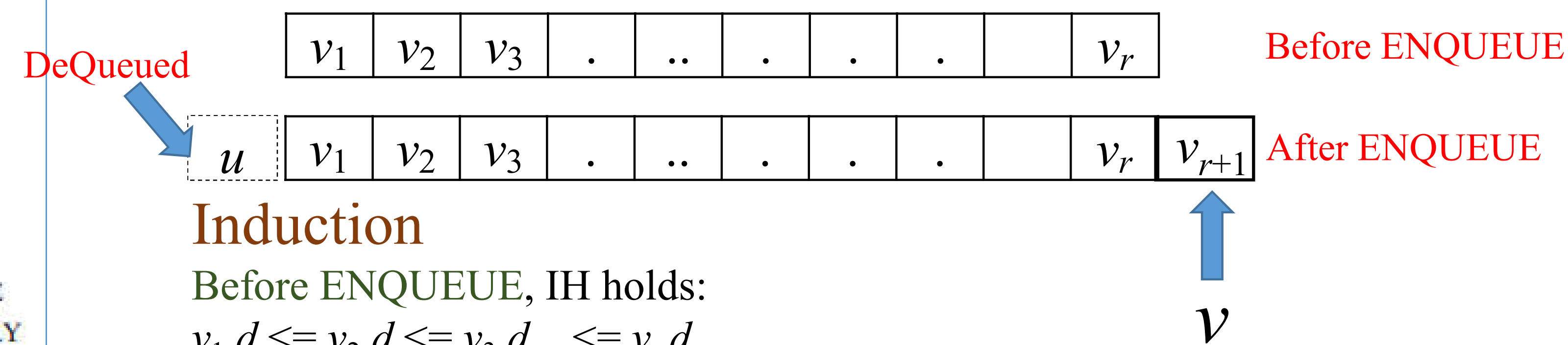
$v_r.d \leq u.d+1 = v.d = v_{r+1}.d$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \leq v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

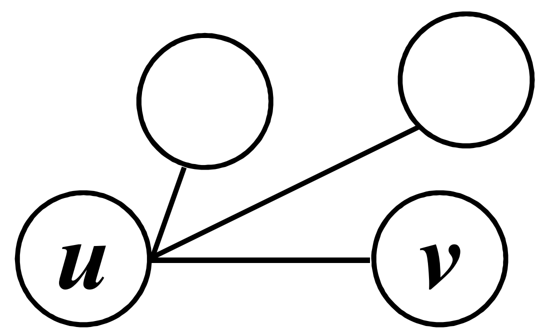


Induction

Before ENQUEUE, IH holds:

$v_1.d <= v_2.d <= v_3.d \ldots <= v_r.d$

$v_r.d <= v_1.d + 1$



After ENQUEUE (of v):

By induction

$v_r.d \leq u.d{+}1 = v.d = v_{r+1}.d$

That means, $v_r.d \leq v_{r+1}.d$

```
BFS(G, s)
for each vertex u ∈ G.V − {s}
    u.color = WHITE
    u.d = ∞
    u.π = NIL
s.color = GRAY
s.d = 0
s.π = NIL
Q = ∅
ENQUEUE(Q, s)
while Q ≠ ∅
    u = DEQUEUE(Q)
    for each v ∈ G.Adj[u]
        if v.color == WHITE
            v.color = GRAY
            v.d = u.d + 1
            v.π = u
            ENQUEUE(Q, v)
    u.color = BLACK
```

Lemma 22.3

Prove: (1) $v_r.d \le v_1.d + 1$

(2) $v_1.d <= v_2.d <= v_3.d <= \cdots <= v_r.d$

Corollary 22.4

Suppose that vertices v_i and v_j are enqueued during the execution of BFS, and that v_i is enqueued before v_j. Then $v_i.d \le v_j.d$ at the time that v_j is enqueued.

v_1	v_2	v_3	.	v_i	..	v_j	.		v_r