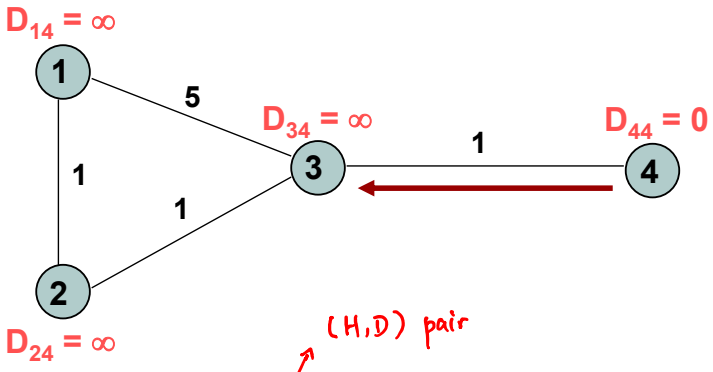
Example: To Destination Node 4



(H,D) pair

	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(-, ∞)	(-, ∞)	(-, ∞)

node 4's DV report

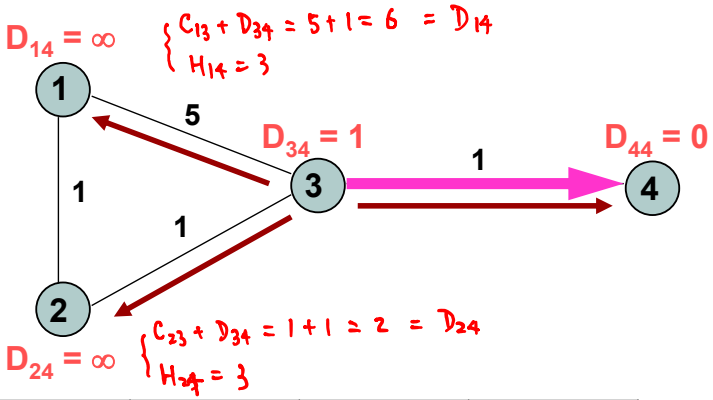
dest j	D_{4j}
1	∞
2	∞
3	∞
4	0 ←

At node 3:

$C_{34} + D_{44} = 1 + 0 = 1$
$\Rightarrow D_{34} = 1$

$H_{34} = 4$

Example: To Destination Node 4

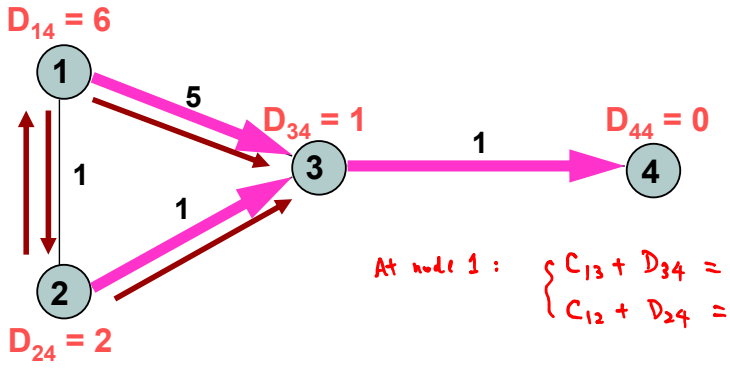


	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(-, ∞)	(-, ∞)	(-, ∞)
	1	(-, ∞)	(-, ∞)	(4, 1)

Node 3's DV report:

dest j	D_{3j}
1	
2	
3	
→ 4	1

Example: To Destination Node 4

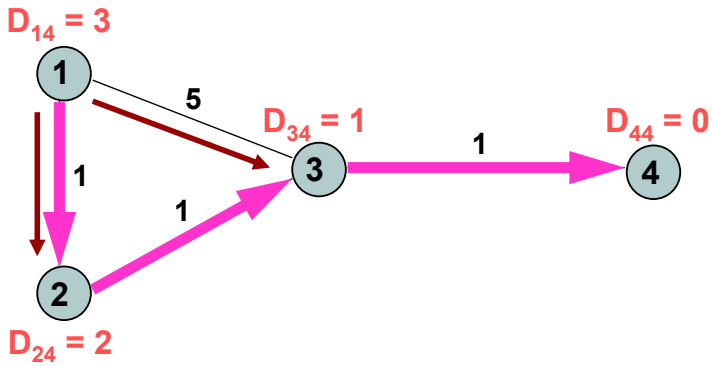


At node 1 : $\begin{cases} C_{13} + D_{34} = 6 \\ C_{12} + D_{24} = 1 + 2 = 3 \end{cases} \Rightarrow D_{14} = 3$

$H_{14} = 2$

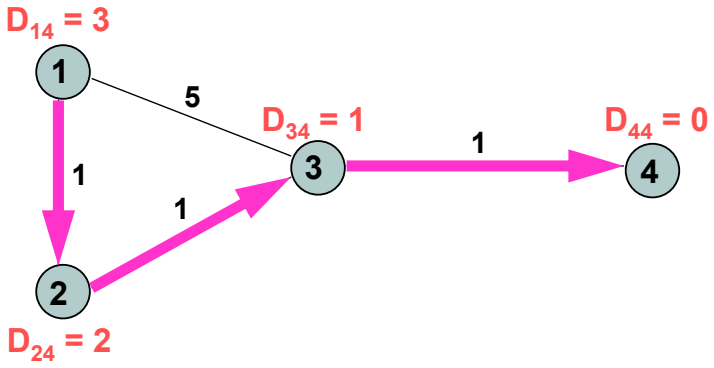
Dest 4	Iteration	Node 1	Node 2	Node 3
	Initial	(-, ∞)	(-, ∞)	(-, ∞)
	1	(-, ∞)	(-, ∞)	(4, 1)
	2	(3, 6)	(3, 2)	(4, 1)

Example: To Destination Node 4



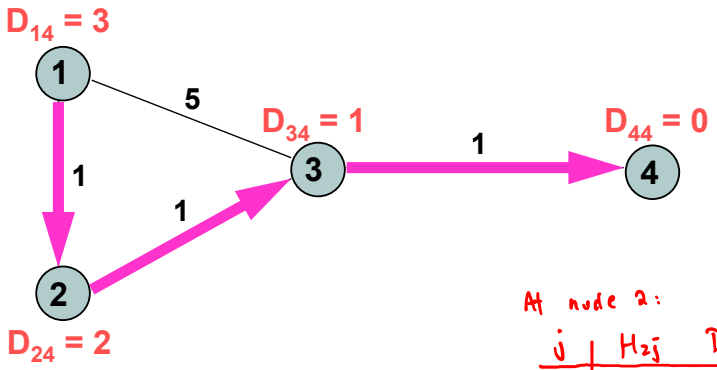
Dest 4	Iteration	Node 1	Node 2	Node 3
	Initial	(-, ∞)	(-, ∞)	(-, ∞)
	1	(-, ∞)	(-, ∞)	(4, 1)
	2	(3, 6)	(3, 2)	(4, 1)
	3	(2, 3)	(3, 2)	(4, 1)

Example: To Destination Node 4



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(-, ∞)	(-, ∞)	(-, ∞)
	1	(-, ∞)	(-, ∞)	(4, 1)
	2	(3, 6)	(3, 2)	(4, 1)
	3	(2, 3)	(3, 2)	(4, 1)
	4	(2, 3)	(3, 2)	(4, 1)

Example: After Protocol Converges, Information at Node 1



Dest J	H_{1j}	D_{1j}	C_{1j}
1	1	0	0
2	2	1	1
3	2	2	5 → ∞
4	2	3	∞

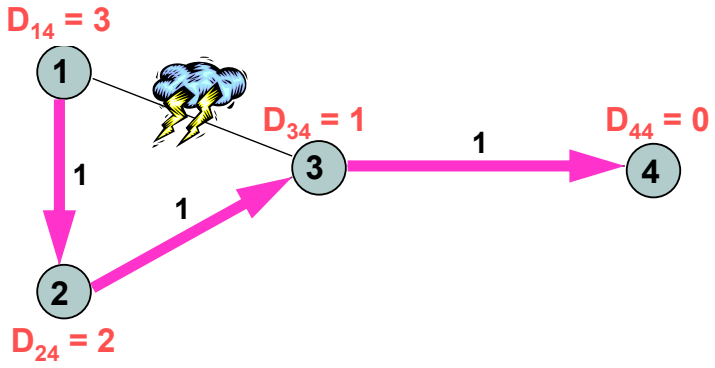
At node 2:

j	H_{2j}	D_{2j}	C_{2j}
1	1	1	1
2	2	0	0
3	3	1	1
4	3	2	∞

At node 3:

j	H_{3j}	D_{3j}	C_{3j}
1	2	2	5
2	2	1	1
3	3	0	0
4	4	1	1

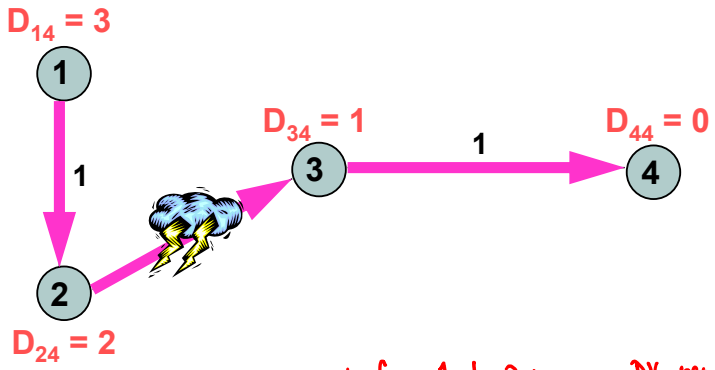
What if link between 1 and 3 breaks?



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(3, 2)	(4, 1)

- local changes may be absorbed locally

Problem Scenario: What if link between 2 and 3 breaks?



At node 2:

	j	H_{2j}	D_{2j}	C_{2j}
	1	1	1	1
	2	2	0	0
→	3	③→1	①→3	①→∞
→	4	③→1	②→4	∞

DV report from 1 to 2:

j	D_{1j}
1	0
2	1
3	2
4	3

DV report from 3 to 2:

j	D_{3j}
1	2
2	1
3	0
4	1

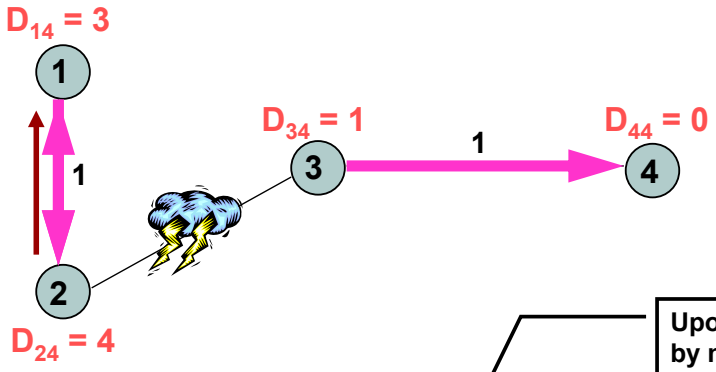
For Dest 4:

$C_{23} + D_{34} = \infty + 1 = \infty$
$C_{21} + D_{14} = 1 + 3 = 4$
} $D_{24} = 4$, $H_{24} = 1$

For Dest 3:

$C_{23} + D_{33} = \infty + 0 = \infty$
$C_{21} + D_{13} = 1 + 2 = 3$
} $D_{23} = 3$, $H_{23} = 1$

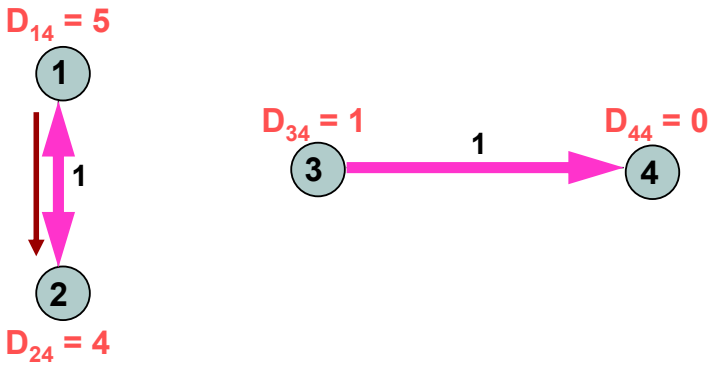
Problem Scenario: What if link between 2 and 3 breaks?



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(1, 4)	(4, 1)

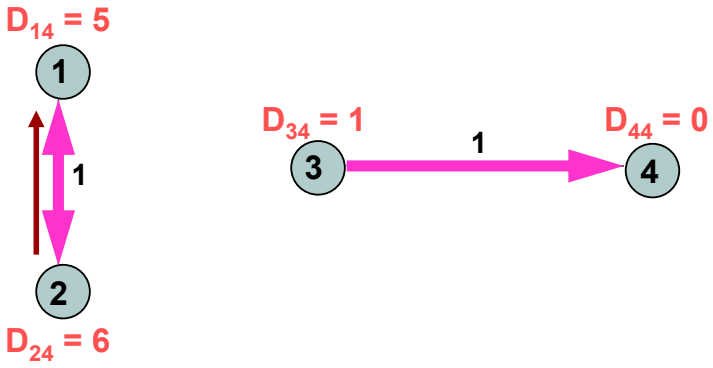
Upon link cost change seen by node 2, it recalculates the shortest path to dest 4, which is through node 1 with distance 4!

Problem Scenario: What if link between 2 and 3 breaks?



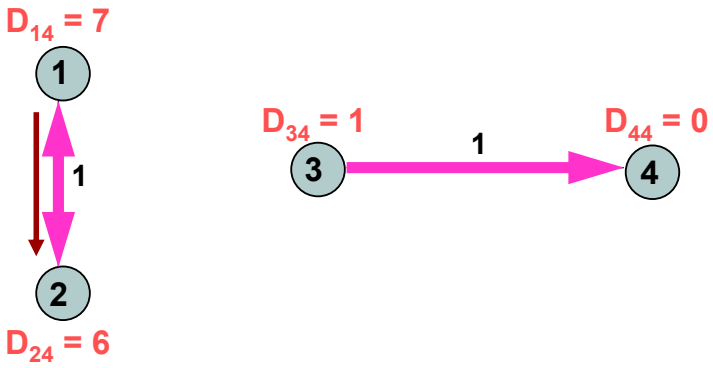
	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(1, 4)	(4, 1)
	1	(2, 5)	(1, 4)	(4, 1)

Problem Scenario: What if link between 2 and 3 breaks?



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(1, 4)	(4, 1)
	1	(2, 5)	(1, 4)	(4, 1)
	2	(2, 5)	(1, 6)	(4, 1)

Problem Scenario: What if link between 2 and 3 breaks?



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(1, 4)	(4, 1)
	1	(2, 5)	(1, 4)	(4, 1)
	2	(2, 5)	(1, 6)	(4, 1)
	3	(2, 7)	(1, 6)	(4, 1)

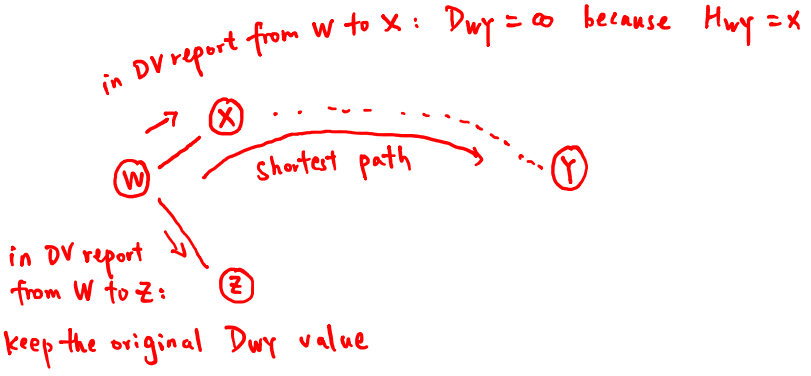
Problem: Routing Loop → Counting to Infinity!

- Causes of Problem
  - Router does not know whether it is in its neighbor's path to a destination
  - Inconsistent routing tables
  - Updates do not reflect reality

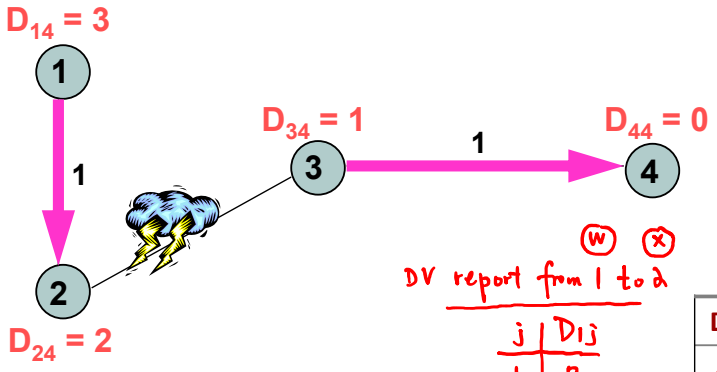
Different views of network situation

Problem: Routing Loop → Counting to Infinity!

- Use heuristics to alleviate the problem
  - Split Horizon with Poisoned Reverse (SHPR)
    - For node W, its neighbor X, and destination Y, if $H_{WY} = X$, then set $D_{WY} = \infty$ in node W's DV report to neighbor X
    - This breaks erroneous direct loops immediately



Example: Problem Solved with SHPR



Dest J	H_{2j}	D_{2j}	C_{2j}
1	1	1	1
2	2	0	0
→ 3	3	1	(1) → ∞
→ 4	3	2	∞

DV report from 1 to 2 (w) (x)

j	D_{1j}
1	0
2	∞
3	∞
4	∞

At node 1:

Dest J	H_{1j}	D_{1j}	C_{1j}
1	1	0	0
2	2	1	1
3	2	2	5
4	2	3	∞

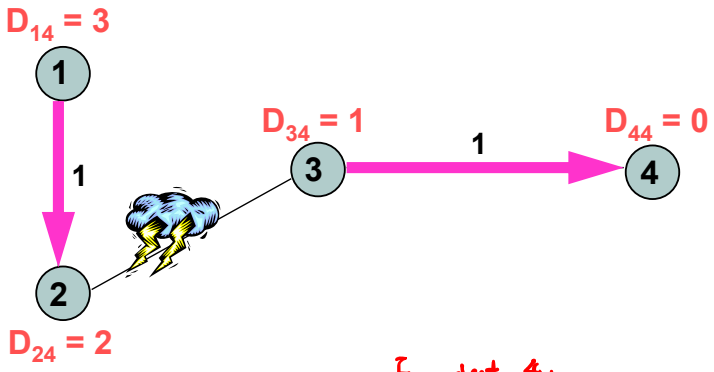
DV report from 3 to 2

j	D_{3j}
1	∞
2	∞
3	0
4	1

At node 3:

Dest J	H_{3j}	D_{3j}	C_{3j}
1	2	2	5
2	2	1	1
3	3	0	0
4	4	1	1

Example: Problem Solved with SHPR



Dest J	H_{2j}	D_{2j}	C_{2j}
1	1	1	1
2	2	0	0
3	~~3~~	1 → ∞	(1) → ∞
4	~~3~~	2 → ∞	∞

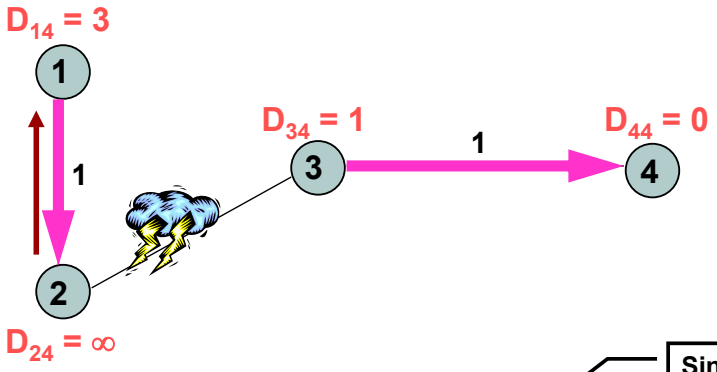
For dest 4:

$C_{21} + D_{14} = 1 + \infty = \infty$

$C_{23} + D_{34} = \infty + 1 = \infty$

$\Rightarrow D_{24} = \infty$

Example: Problem Solved with SHPR



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(-, ∞)	(4, 1)

Since node 1 advertised to node 2 that its minimum cost to dest 4 is ∞, node 2 finds no route to dest 4.

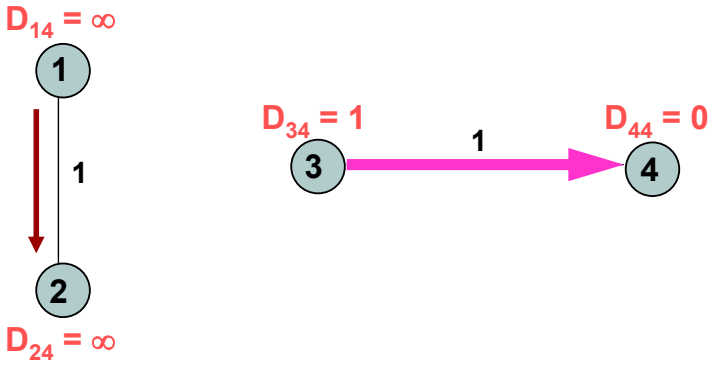
$D_{24} = \infty$ in DV report from 2 to 1 :

At 1: $C_{12} + D_{24} = 1 + \infty = \infty$

$\Rightarrow D_{14} = \infty$

j	D_{2j}
1	∞
2	0
3	∞
4	∞

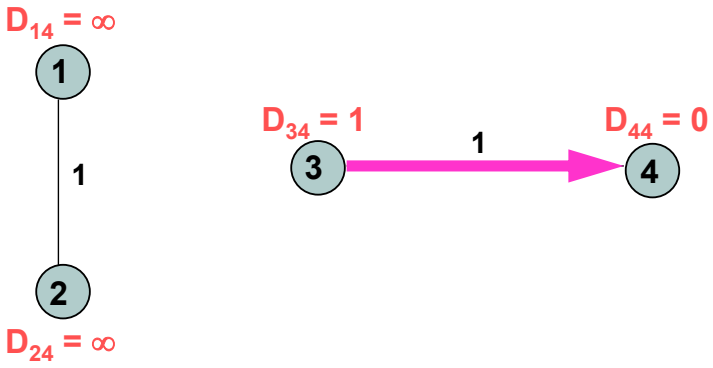
Example: Problem Solved with SHPR



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(-, ∞)	(4, 1)
	1	(-, ∞)	(-, ∞)	(4, 1)

Node 1 also finds no route to dest 4.

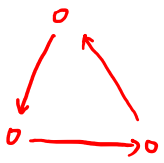
Example: Problem Solved with SHPR



	Iteration	Node 1	Node 2	Node 3
Dest 4	Initial	(2, 3)	(3, 2)	(4, 1)
	After break	(2, 3)	(-, ∞)	(4, 1)
	1	(-, ∞)	(-, ∞)	(4, 1)
	2	(-, ∞)	(-, ∞)	(4, 1)

SHPR is NOT a Loop-Free Solution!

- SHPR eliminates the routing loops that only involve 2 nodes
- SHPR does not eliminate the routing loops that involve >2 nodes



SHPR is NOT a Loop-Free Solution!

- Example Loop-free Scheme: Path Vector Routing
  - Each node sends to its neighbors the entire path information to every destination
  - Each node uses a neighbor's information for a certain destination only if itself is not on this neighbor's path to the destination
  - Each node prepends itself to paths before further propagation
  - Example:
    - Node 1's path vector

Dest	1	2	3	4
Distance	0	1	2	3
Path	<1>	<1, 2>	<1, 2, 3>	<1, 2, 3, 4>

RIP (Routing Information Protocol)

- Distance Vector Routing Protocol
  - Split Horizon with Poisoned Reverse
- Runs on top of UDP, port # 520, **"routed"** BSD Unix program
- Routing Metric: number of hops
- Max number of hops is limited to 15
  - Suitable for small networks (local area environments)
  - 16 is reserved to represent infinity
  - Small number helps to limit the Counting-to-Infinity Problem

BGP (Border Gateway Protocol)

- Path Vector Routing Protocol
  - Avoid routing loops
- BGP
  - Is a reachability protocol
  - Uses TCP to send updates
    - Reliable transmission
    - Allow incremental updates
  - Allows for policy routing
    - Path selection by policy rather than path optimality