COMP 3331/9331: Computer Networks and Applications

Assignment Project Exam Help
https://tutorcs.com

Week 8

WeChat: cstutorcs

Network Layer: Data Plane + Control Plane (Routing)

Chapter 4: Section 4.3

Chapter 5: Section 5.1 – 5.2, 5.6

Network Layer, data plane: outline

4.1 Overview of Network layer

- data plane
- control plane

Assignment Project Exam Help

4.2 What's inside a router

https://tutorcs.com

4.3 IP: Internet Protocol

- datagram format

WeChat: cstutorcs

- fragmentation
- IPv4 addressing
- network address translation
- IPv6

IPv6: motivation

- *initial motivation:* 32-bit address space soon to be completely allocated.
- additional motivation:
  - header format helps speed processing/forwarding
  - header changes to facilitate QoS

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

IPv6 datagram format:

- fixed-length 40-byte header
- no fragmentation allowed

https://www.google.com/intl/en/ipv6/statistics.html

IPv6 datagram format

priority: identify priority among datagrams in flow (traffic class)

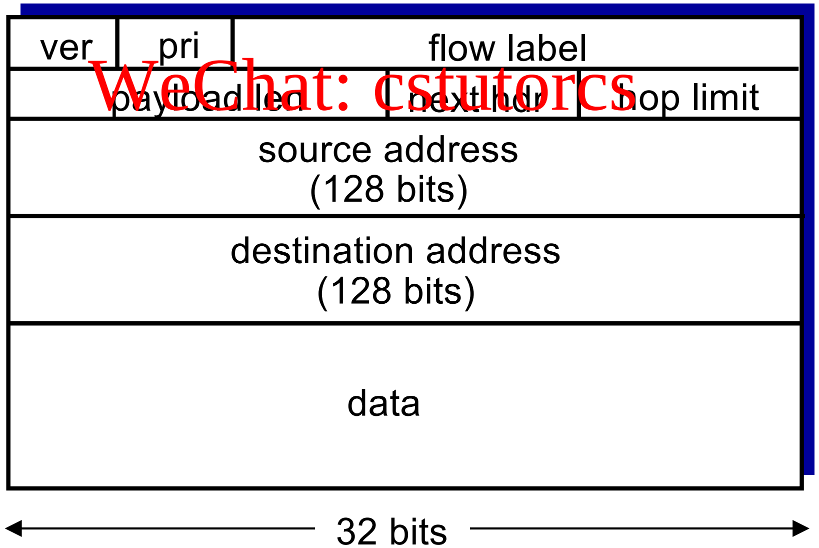
flow Label: identify datagrams in same "flow."

(concept of "flow" not well defined).

Assignment Project Exam Help

next header: identify upper layer protocol for data

https://tutorcs.com



WeChat: cstutorcs

Other changes from IPv4

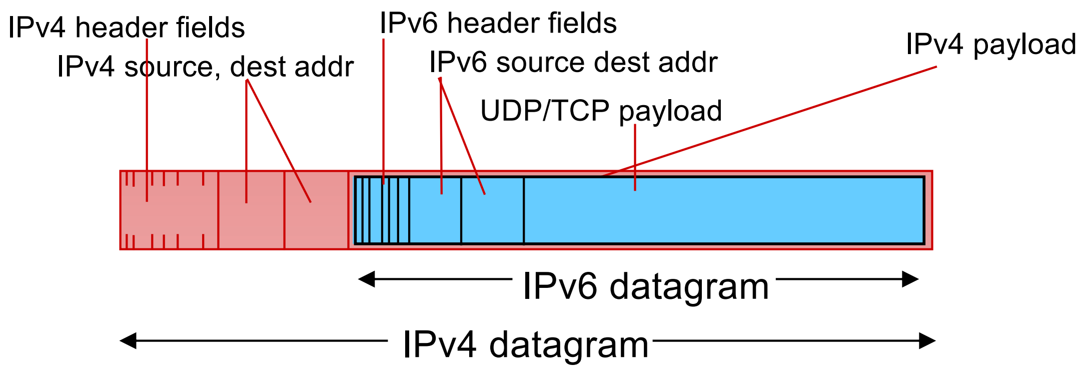
- *checksum:* removed entirely to reduce processing time at each hop
- *options:* allowed, but outside of header, indicated by "Next Header" field
- *ICMPv6:* new version of ICMP
  - additional message types, e.g., "Packet Too Big"
  - multicast group management functions

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

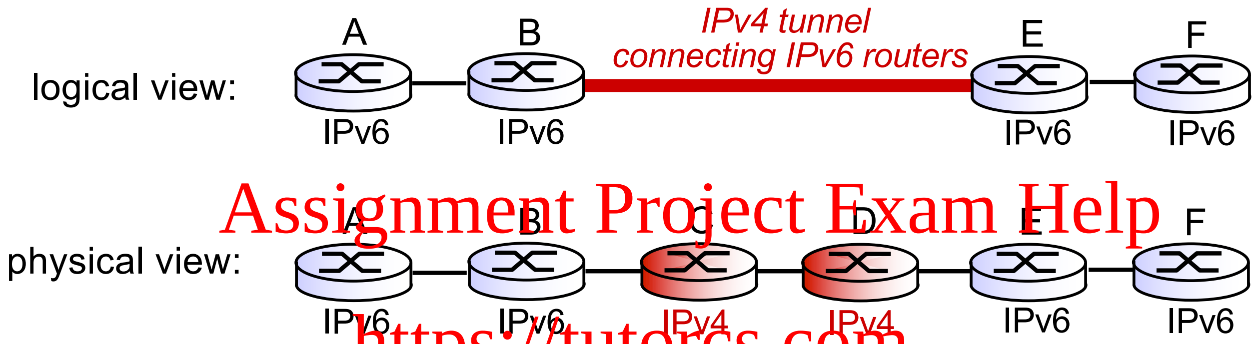
Transition from IPv4 to IPv6

- not all routers can be upgraded simultaneously
  - no "flag days"
  - how will network operate with mixed IPv4 and IPv6 routers?
- *tunneling:* IPv6 datagram carried as *payload* in IPv4 datagram among IPv4 routers

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

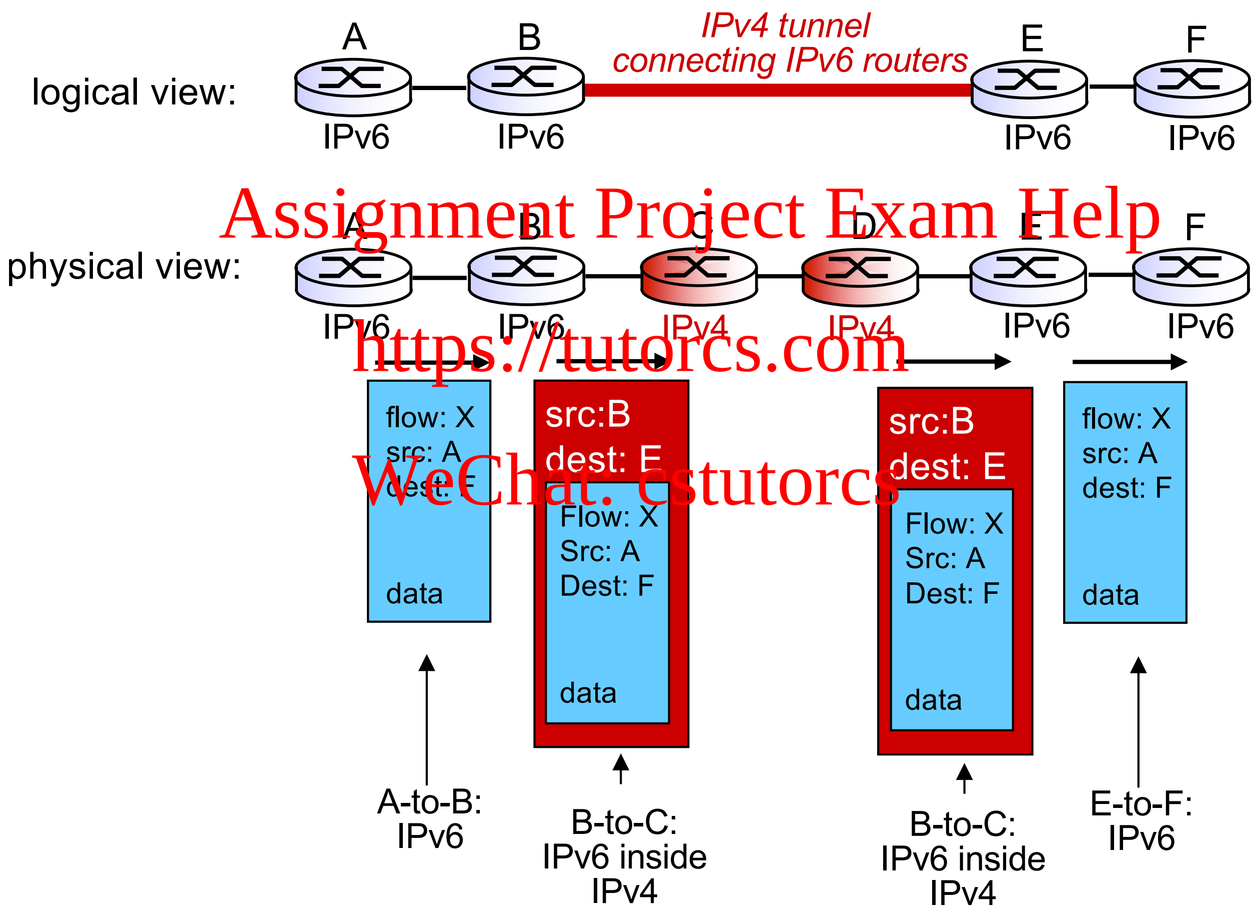


Tunneling (IPv6 over IPv4)



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

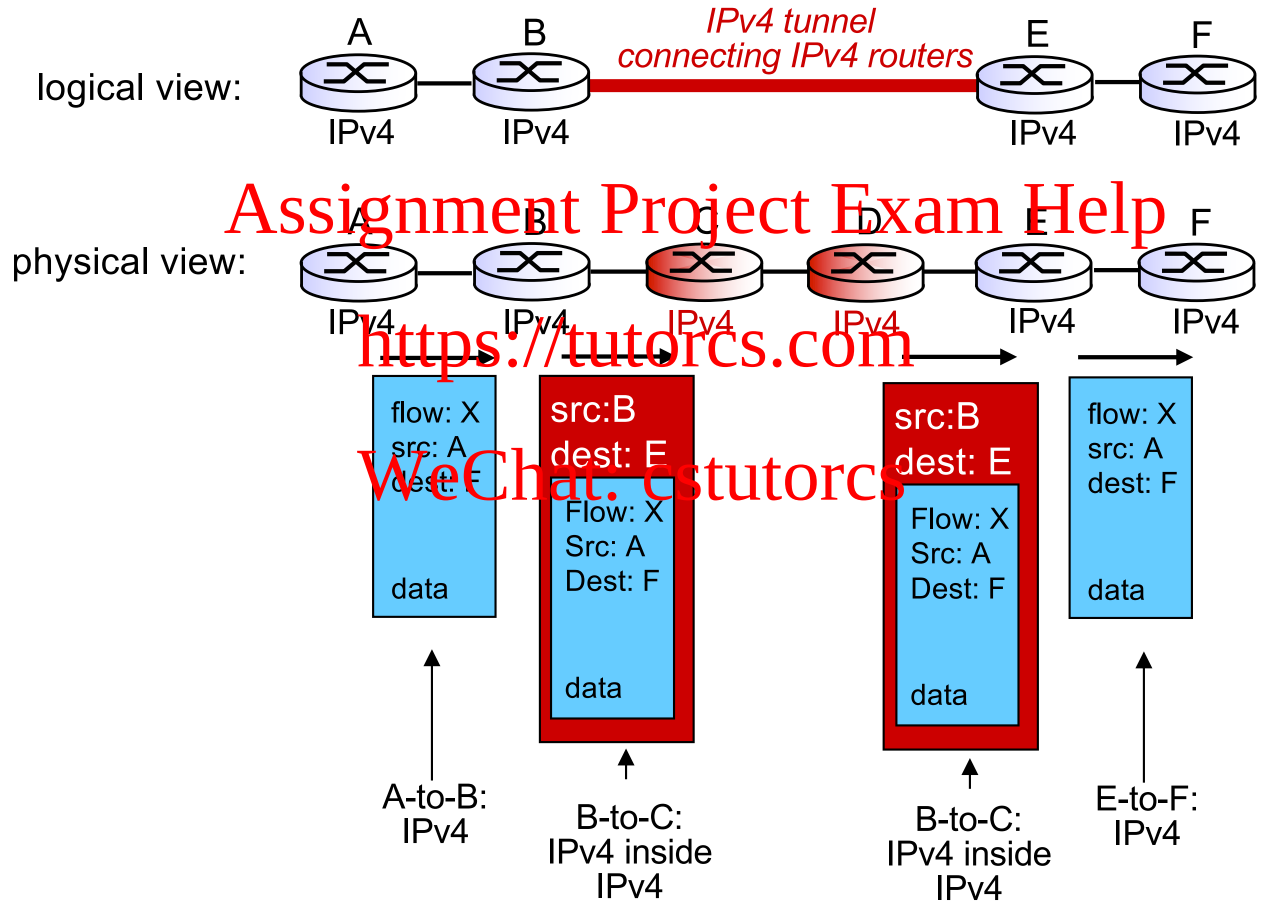
Tunneling (IPv6 over IPv4)



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Tunneling (IPv4 over IPv4)

Used in VPNs



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Network layer, control plane: outline

5.1 introduction

5.2 routing protocols

- link state
- distance vector
- hierarchical routing

5.6 ICMP: The Internet Control Message Protocol

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Self study (not on exam)

Network-layer functions

Recall: two network-layer functions:

- *forwarding:* move packets from router's input to appropriate router output

data plane

- *routing:* determine route taken by packets from source to destination

control plane

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Two approaches to structuring network control plane:

- per-router control (traditional)
- logically centralized control (software defined networking)

Per-router control plane

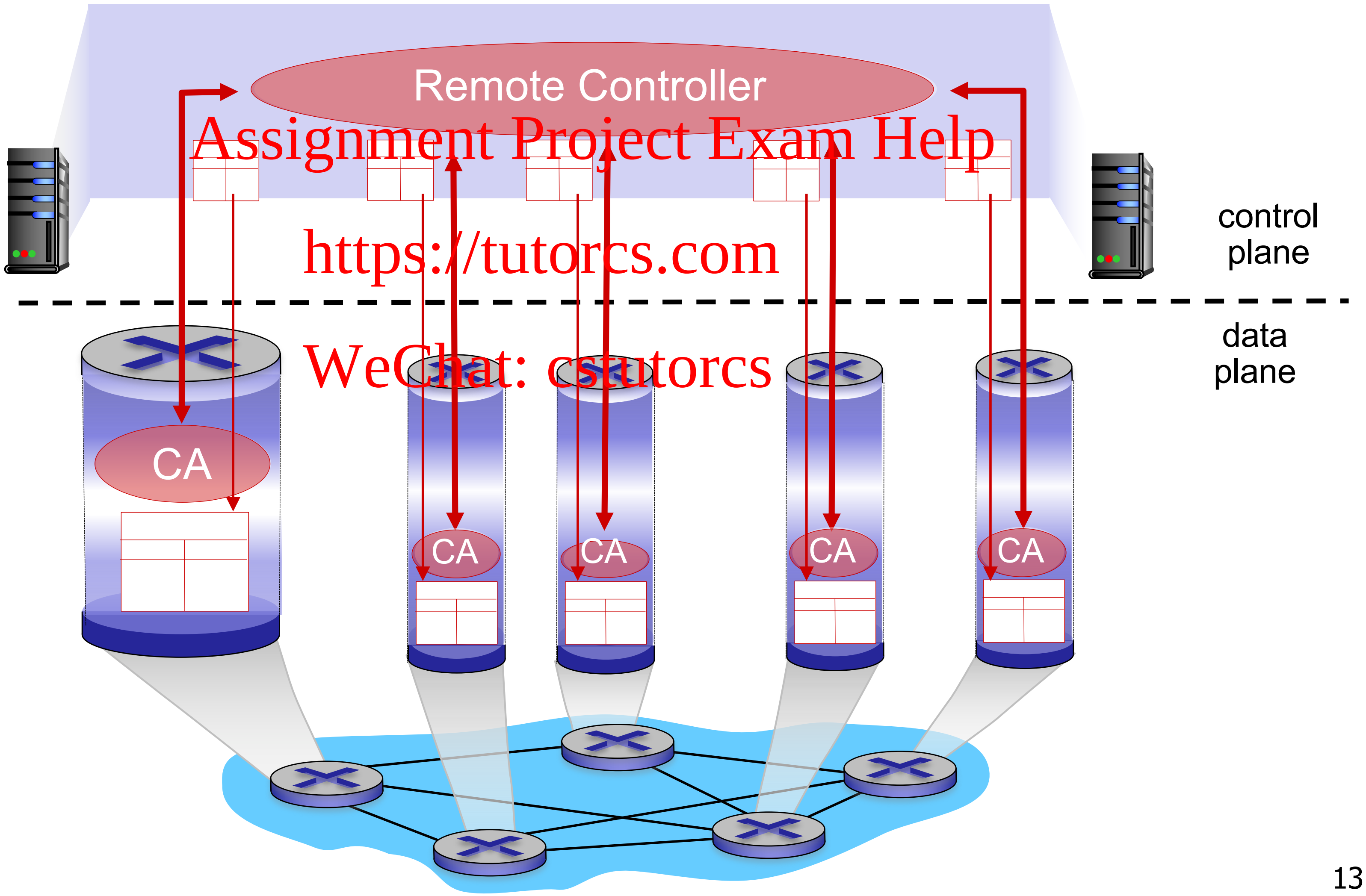
Individual routing algorithm components *in each and every router* interact with each other in control plane to compute forwarding tables

Assignment Project Exam Help

https://tutorcs.com
WeChat: cstutorcs

Logically centralized control plane

A distinct (typically remote) controller interacts with local control agents (CAs) in routers to compute forwarding tables



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Network layer, control plane: outline

5.1 introduction

5.2 routing protocols

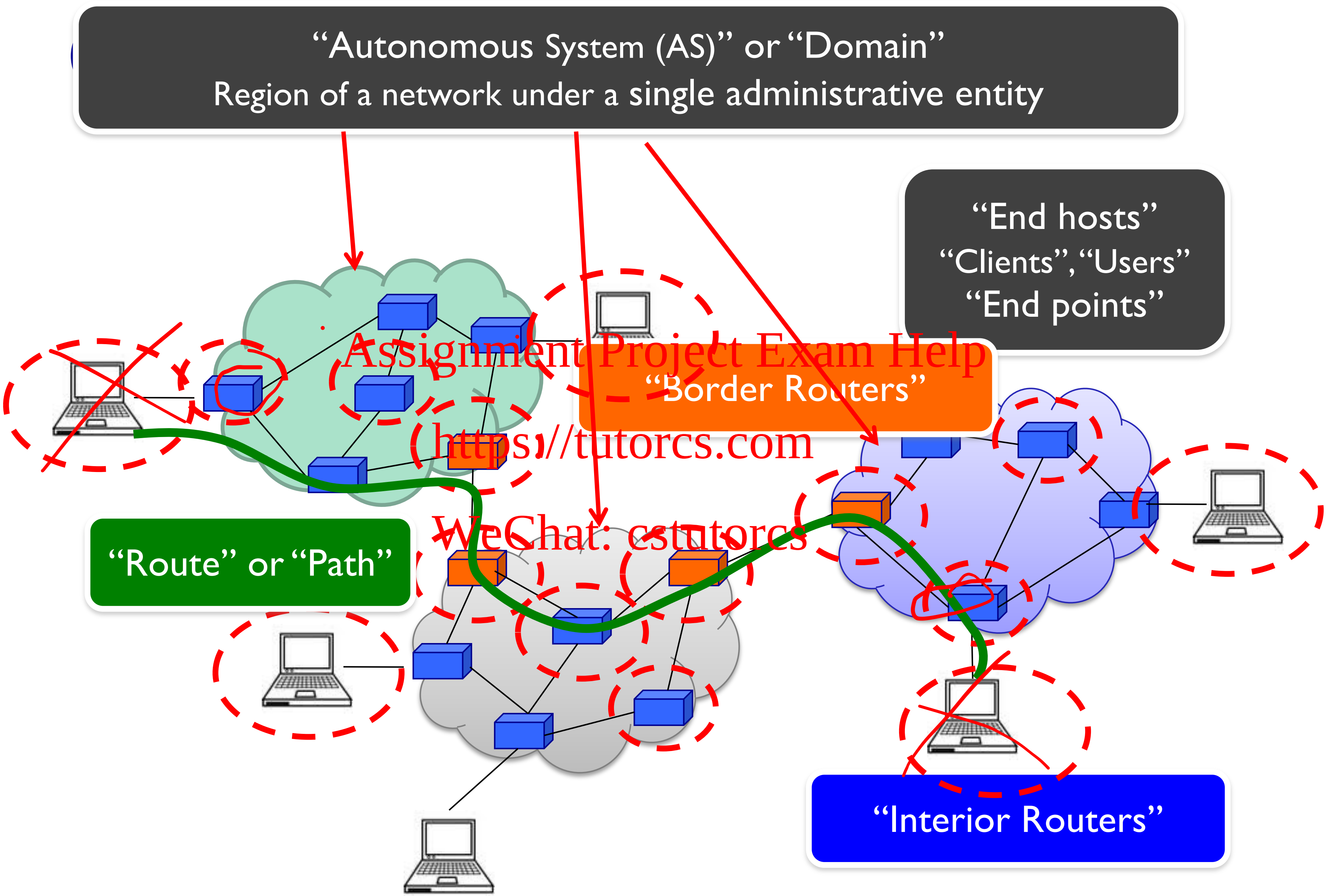
- link state
- distance vector
- Hierarchical routing

5.6 ICMP: The Internet Control Message Protocol

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Internet Routing

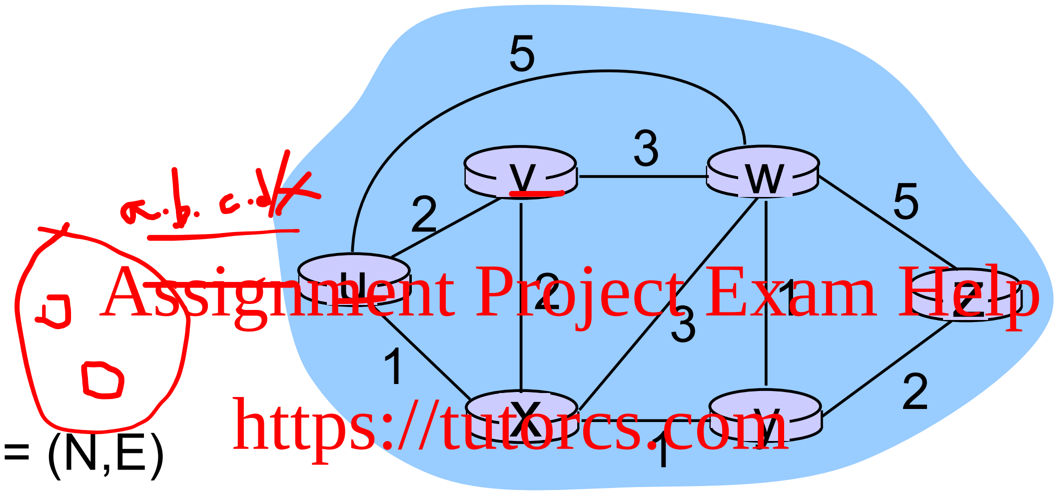
- Internet Routing works at two levels

- Each AS runs an intra-domain routing protocol that establishes routes within its domain
  - AS -- region of network under a single administrative entity
  - Link State, e.g., Open Shortest Path First (OSPF)
  - Distance Vector, e.g., Routing Information Protocol (RIP)

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

- ASes participate in an inter-domain routing protocol that establishes routes between domains
  - Path Vector, e.g., Border Gateway Protocol (BGP)

Graph abstraction



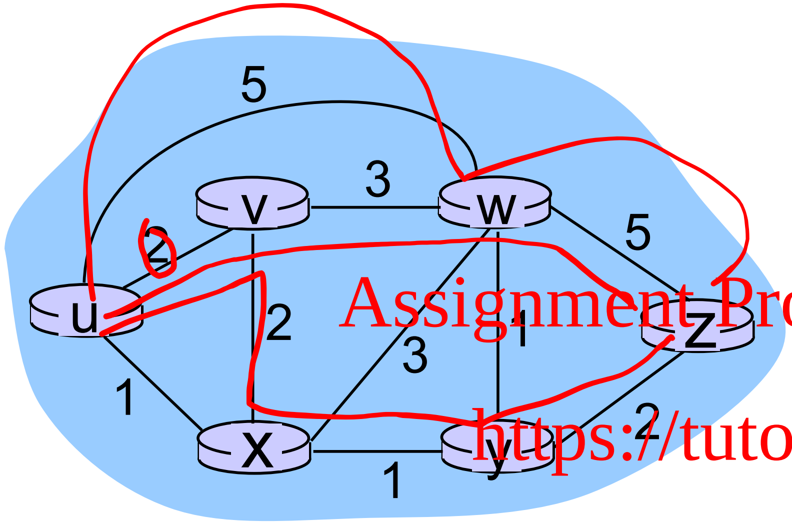
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

graph: G = (N,E)

N = set of routers = { u, v, w, x, y, z }

E = set of links ={ (u,v), (u,x), (u,w), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) }

Graph abstraction: costs



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

c(x,x') = cost of link (x,x')
e.g., c(w,z) = 5

cost of path $(x_1, x_2, x_3,\ldots, x_p) = c(x_1,x_2) + c(x_2,x_3) + \ldots + c(x_{p-1},x_p)$

key question: what is the least-cost path between u and z ?
routing algorithm: algorithm that finds that least cost path

Link Cost

- Typically simple: all links are equal
- Least-cost paths => shortest paths (hop count)
- Network operators add policy exceptions
  - Lower operational costs
  - Peering agreements
  - Security concerns

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Network layer, control plane: outline

5.1 introduction

5.2 routing protocols

- link state
- distance vector
- hierarchical routing

5.6 ICMP: The Internet Control Message Protocol

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Routing algorithm classes

Link State (Global)

- Routers maintain cost of each link in the network
- Connectivity/cost changes flooded to all routers
- Converges quickly (less inconsistency, looping, etc.)
- Limited network sizes

Distance Vector (Decentralised)

- Routers maintain next hop & cost of each destination.
- Connectivity/cost changes iteratively propagate form neighbour to neighbour
- Requires multiple rounds to converge
- Scales to large networks

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

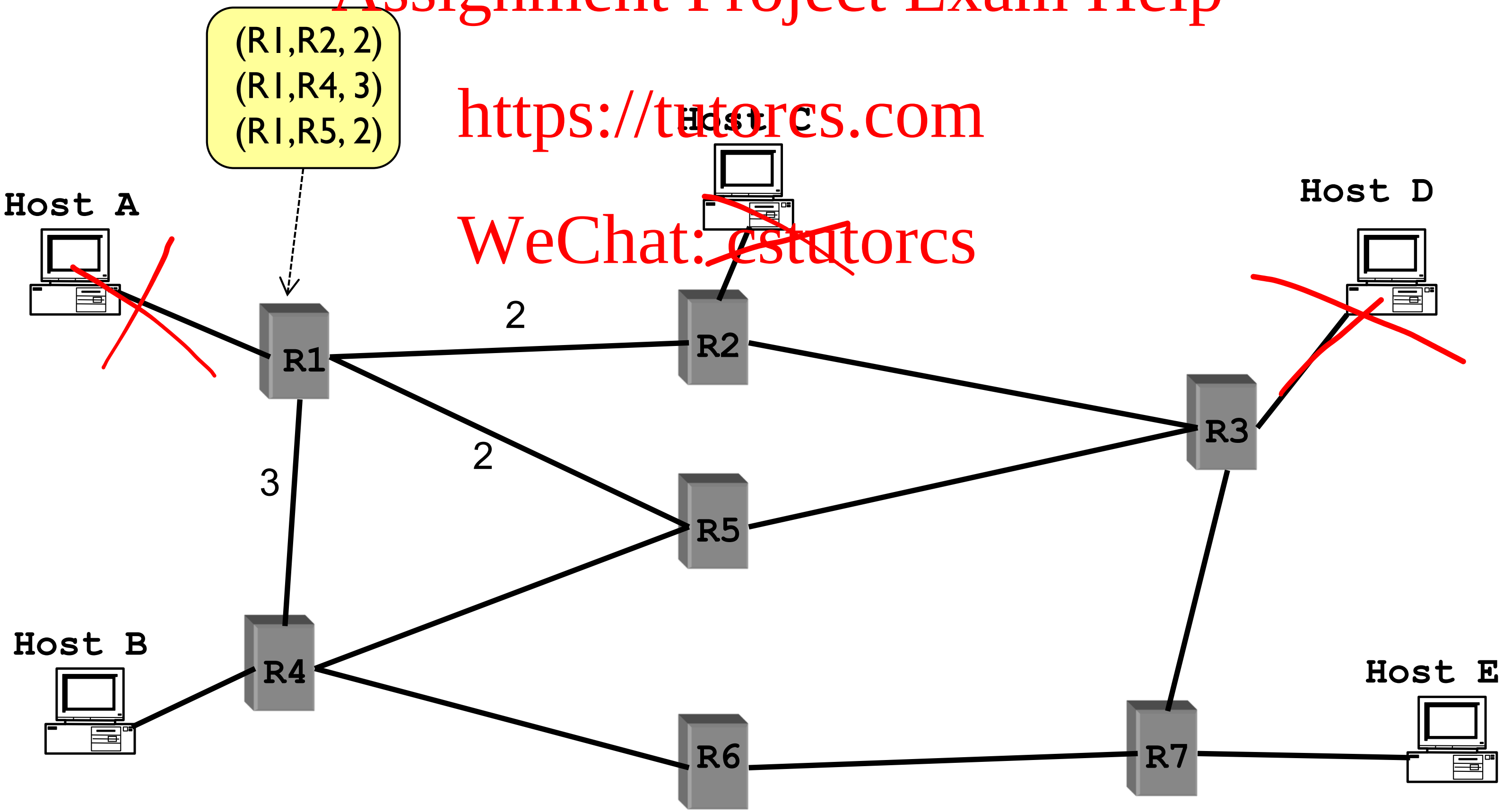
Link State Routing

- Each node maintains its local "link state" (LS)
  - i.e., a list of its directly attached links and their costs

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



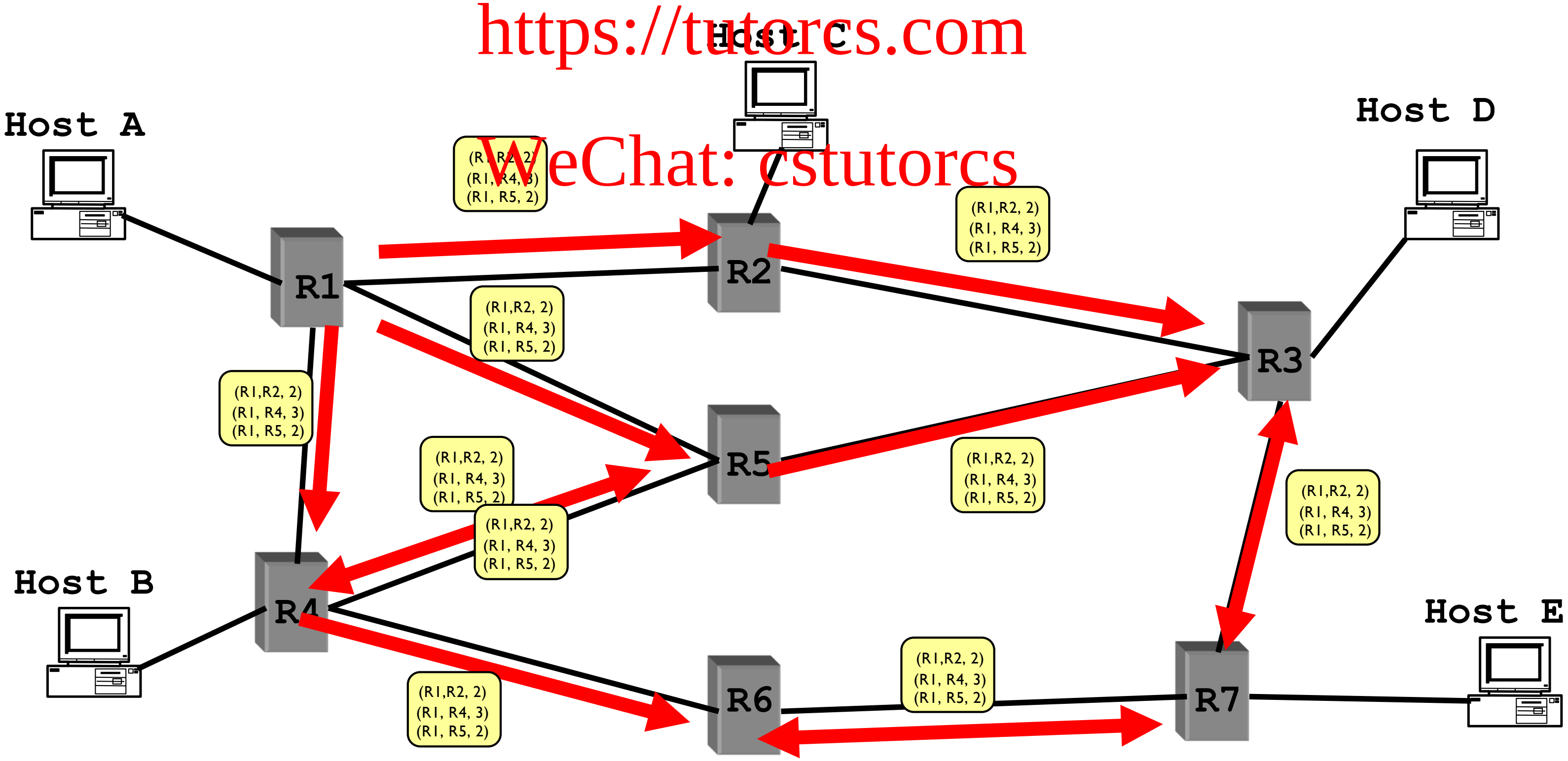
Link State Routing

- Each node maintains its local "link state" (LS)
- Each node floods its local link state
  - on receiving a new LS message, a router forwards the message to all its neighbors other than the one it received the message from

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



Flooding LSAs

- Routers transmit Link State Advertisement (LSA) on links
  - A neighbouring router forwards out on all links except incoming
  - Keep a copy locally; don't forward previously-seen LSAs
- Challenges
  - Packet loss
  - Out of order arrival
- Solutions
  - Acknowledgements and retransmissions
  - Sequence numbers
  - Time-to-live for each packet

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

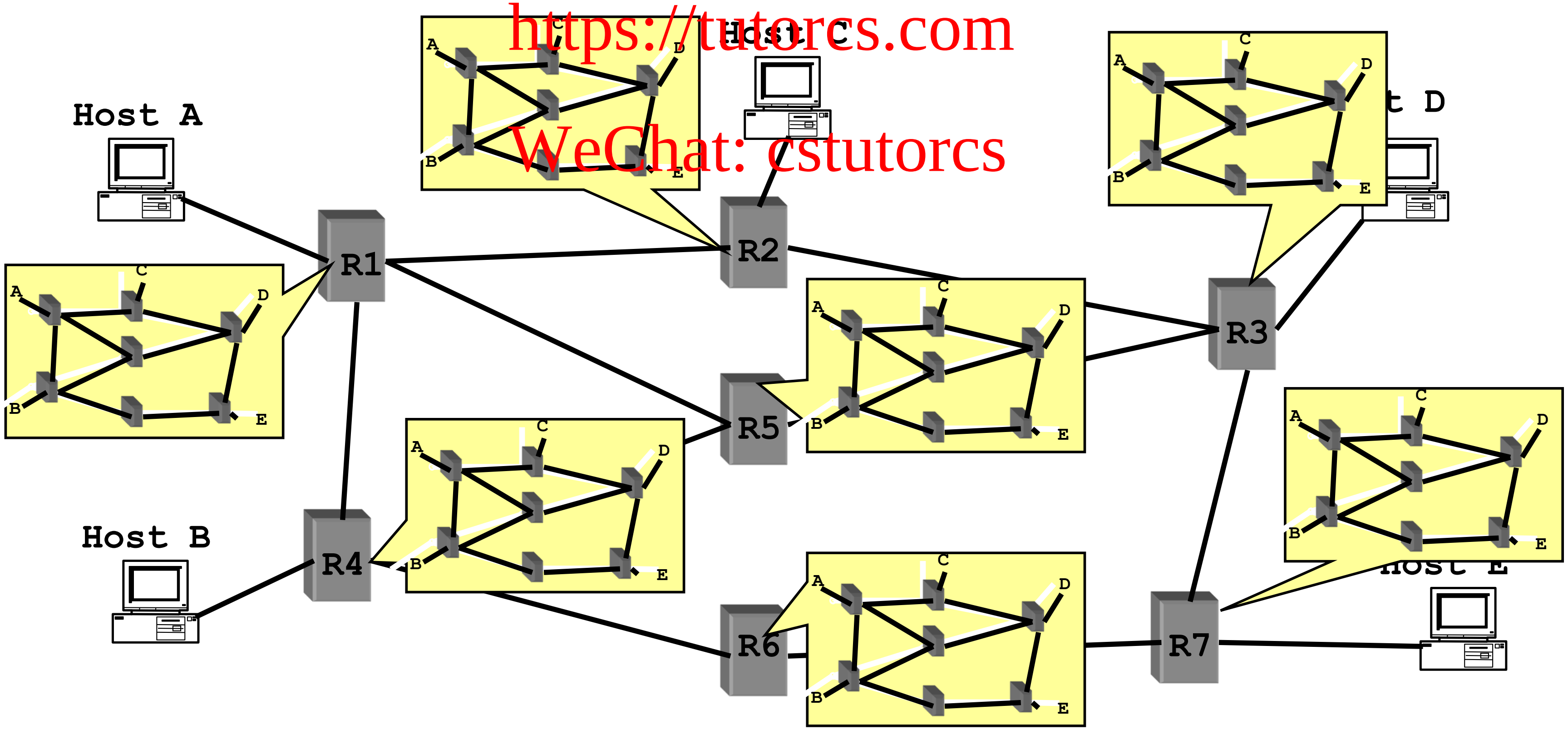
Link State Routing

- Each node maintains its local "link state" (LS)
- Each node floods its local link state
- Eventually, each node learns the entire network topology
  - Can use Dijkstra's to compute the shortest paths between nodes

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



A Link-State Routing Algorithm

Dijkstra's algorithm

- net topology, link costs known to all nodes
  - accomplished via "link state broadcast"
  - all nodes have same info
- computes least cost paths from one node ("source") to all other nodes
  - gives *forwarding table* for that node
- iterative: after k iterations, know least cost path to k dest.'s

notation:

- c(x,y): link cost from node x to y; = ∞ if not direct neighbors
- D(v): current value of cost of path from source to dest. v
- p(v): predecessor node along path from source to v
- N': set of nodes whose least cost path definitively known

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Dijsktra's Algorithm

```
Initialization:
  N' = {u}
  for all nodes v
    if v adjacent to u
      then D(v) = c(u,v)
    else D(v) = ∞

Loop
  find w not in N' such that D(w) is a minimum
  add w to N'
  update D(v) for all v adjacent to w and not in N' :
     D(v) = min( D(v), D(w) + c(w,v) )
  /* new cost to v is either old cost to v or known
   shortest path cost to w plus cost from w to v */
until all nodes in N'
```

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

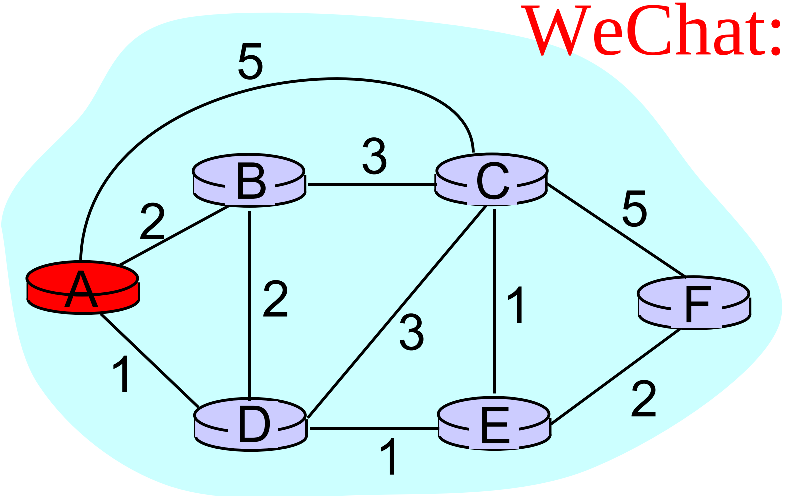
Example: Dijkstra’s Algorithm

Step	Set N’	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1						
2						
3						
4						
5						

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs




```
Initialization:
  N’ = {A};
  for all nodes v
    if v adjacent to A
      then D(v) = c(A,v);
      else D(v) = ∞;
…
```

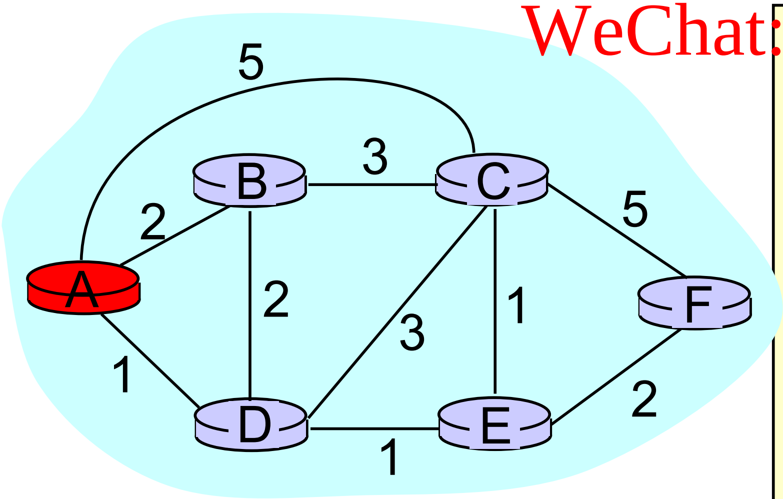
Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1						
2						
3						
4						
5						

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs




```
...
Loop
    find w not in N' s.t. D(w) is a minimum;
    add w to N';
   update D(v) for all v adjacent
      to w and not in N':
   If D(w) + c(w,v) < D(v) then
      D(v) = D(w) + c(w,v); p(v) = w;
    until all nodes in N';
```

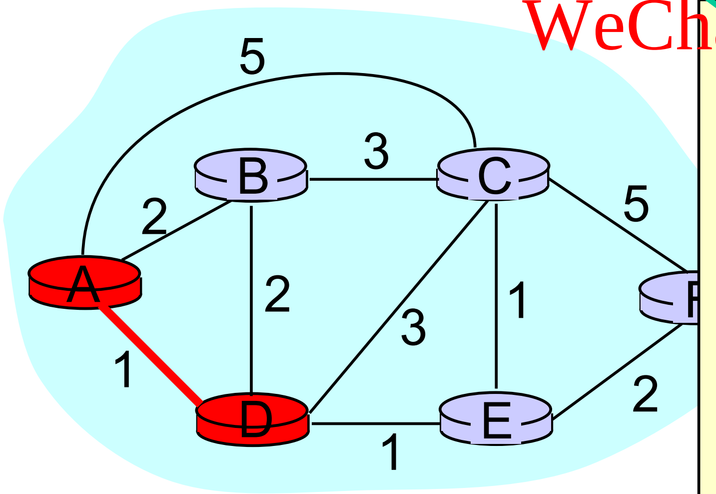
Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD					
2						
3						
4						
5						

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs




```
...
Loop
   find w not in N' s.t. D(w) is a minimum;
   add w to N';
   update D(v) for all v adjacent
     to w and not in N':
   If D(w) + c(w,v) < D(v) then
      D(v) = D(w) + c(w,v); p(v) = w;
 until all nodes in N';
```

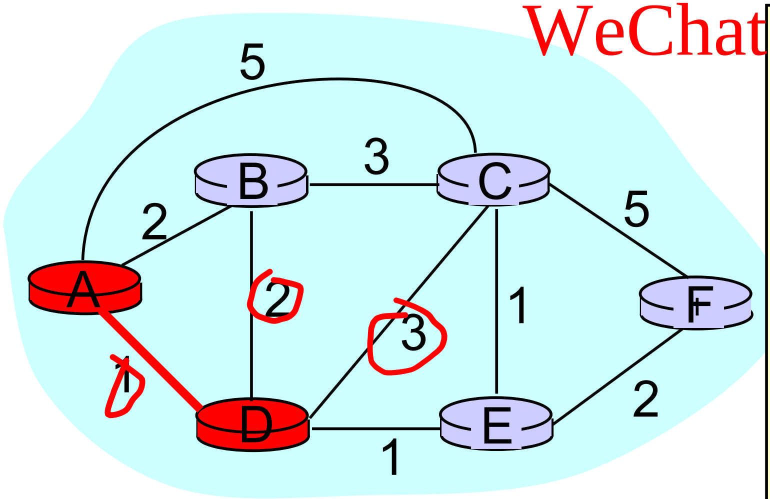
Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD	2,A	4,D		2,D	
2						
3						
4						
5						

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

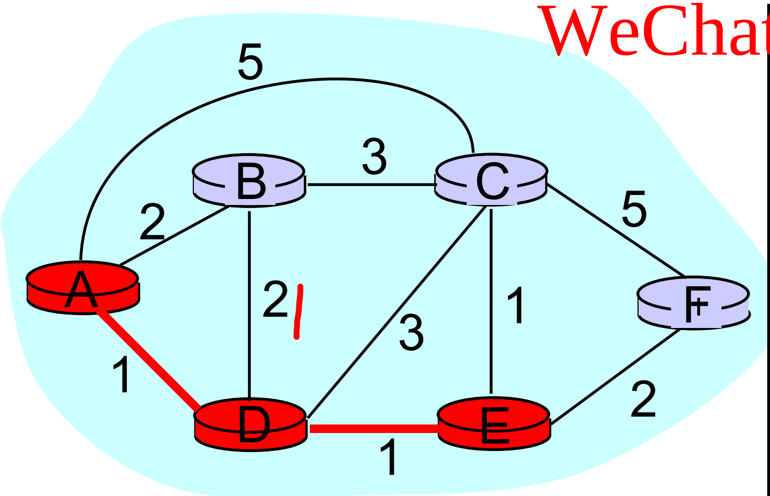


Loop
    find **w** not in **N'** s.t. D(w) is a minimum;
    add **w** to **N'**;
    update D(v) for all **v** adjacent to **w** and not in **N'**:
    If D(w) + c(w,v) < D(v) then
        D(v) = D(w) + c(w,v); p(v) = w;
until all nodes in N';

Example: Dijkstra’s Algorithm

Step	Set N’	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD	2, A	4,D		2,D	
2	ADE	2,A	3,E			4,E
3						
4						
5						

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

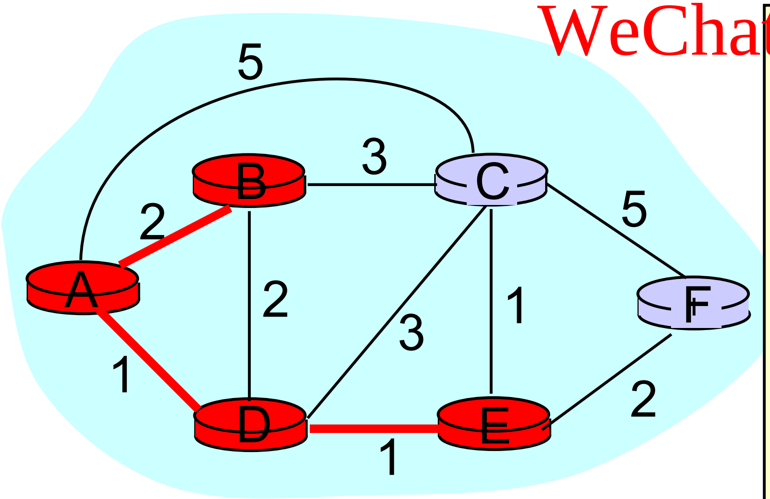



```
Loop
   find w not in N’ s.t. D(w) is a minimum;
   add w to N’;
  update D(v) for all v adjacent
     to w and not in N’:
  If D(w) + c(w,v) < D(v) then
     D(v) = D(w) + c(w,v); p(v) = w;
 until all nodes in N’;
```

Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD	2,A	4,D		2,D	
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4						
5						

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

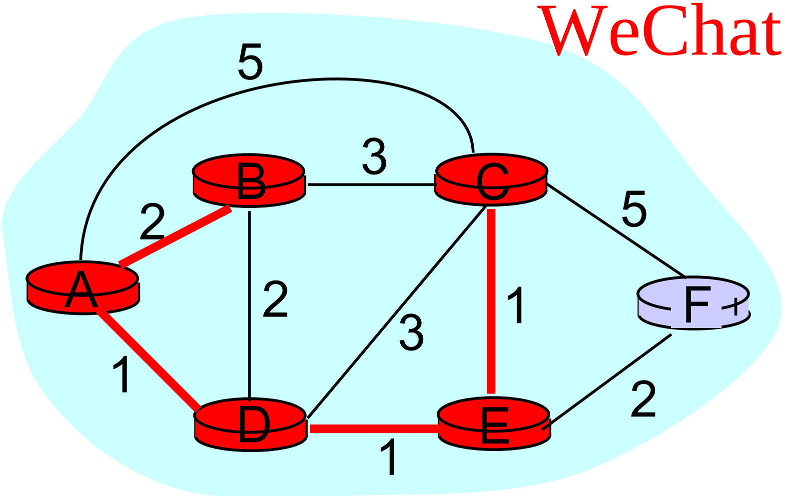


Loop
  find **w** not in **N'** s.t. D(w) is a minimum;
  add **w** to **N'**;
  update D(v) for all **v** adjacent
    to **w** and not in **N'**:
  If D(w) + c(w,v) < D(v) then
    D(v) = D(w) + c(w,v); p(v) = w;
until all nodes in N';

Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD	2,A	4,D		2,D	
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5						

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs




```
...
Loop
    find w not in N' s.t. D(w) is a minimum;
    add w to N';
   update D(v) for all v adjacent
      to w and not in N':
   If D(w) + c(w,v) < D(v) then
      D(v) = D(w) + c(w,v); p(v) = w;
 until all nodes in N';
```

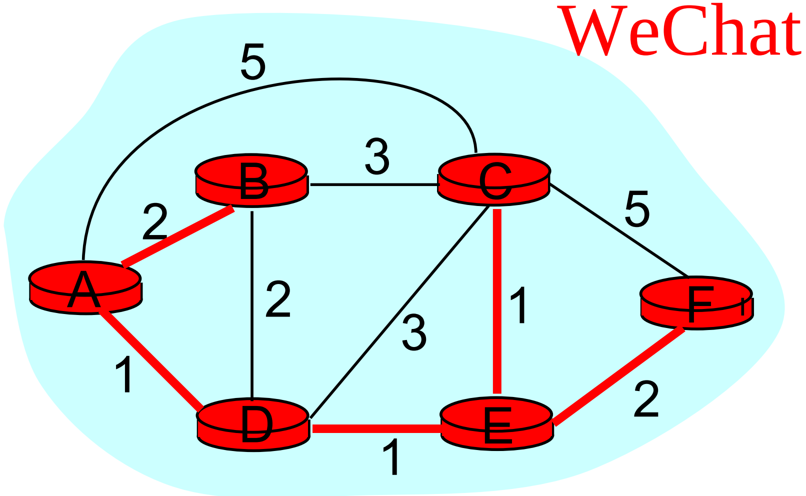
Example: Dijkstra's Algorithm

Step	Set N'	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD	2,A	4,D		2,D	
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5	ADEBCF					

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs




```
Loop
    find w not in N' s.t. D(w) is a minimum;
    add w to N';
   update D(v) for all v adjacent
      to w and not in N':
   If D(w) + c(w,v) < D(v) then
      D(v) = D(w) + c(w,v); p(v) = w;
   until all nodes in N';
```

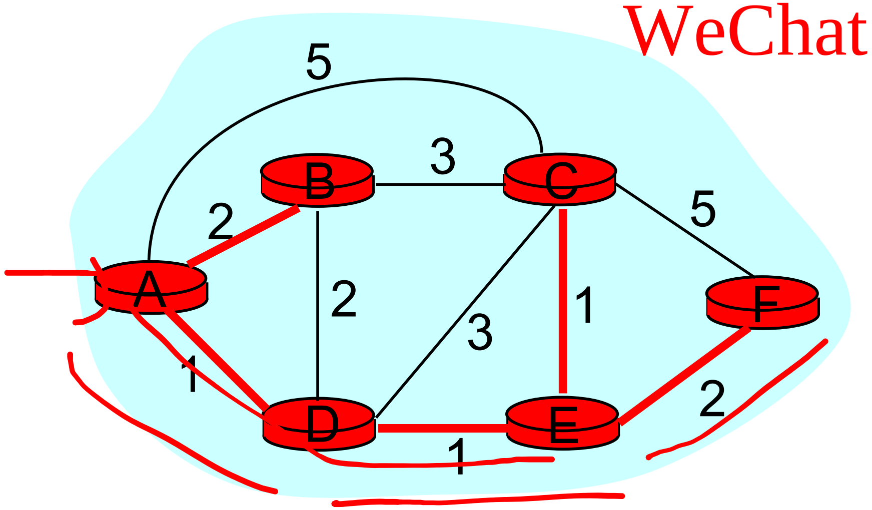
Example: Dijkstra’s Algorithm

Step	Set N’	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	∞	∞
1	AD		4,D		2,D	
2	ADE		3,E			4,E
3	ADEB					
4	ADEBC					
5	ADEBCF					

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



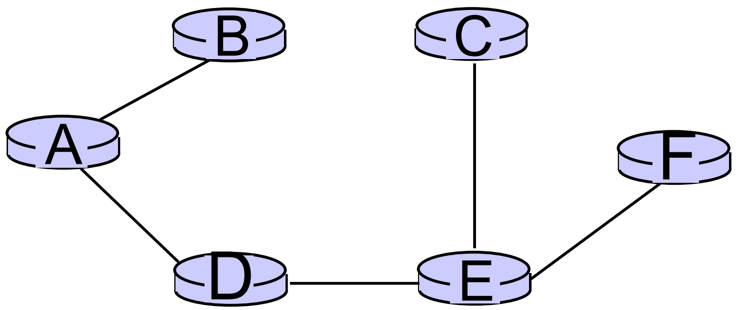
To determine path A → C (say), work backward from C via p(v)

The Forwarding Table

- Running Dijkstra at node A gives the shortest path from A to all destinations
- We then construct the *forwarding table*

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

resulting shortest-path tree from A:



Destination	Link Next h
B	(A,B)
C	(A,D)
D	(A,D)
E	(A,D)
F	(A,D)

Issue #1: Scalability

- How many messages needed to flood link state messages?
  - O(N x E), where N is #nodes; E is #edges in graph

Assignment Project Exam Help

- Processing complexity for Dijkstra's algorithm?
  - O(N^2), because we check all nodes w not in N' at each iteration and we have O(N) iterations

https://tutorcs.com

WeChat: cstutorcs

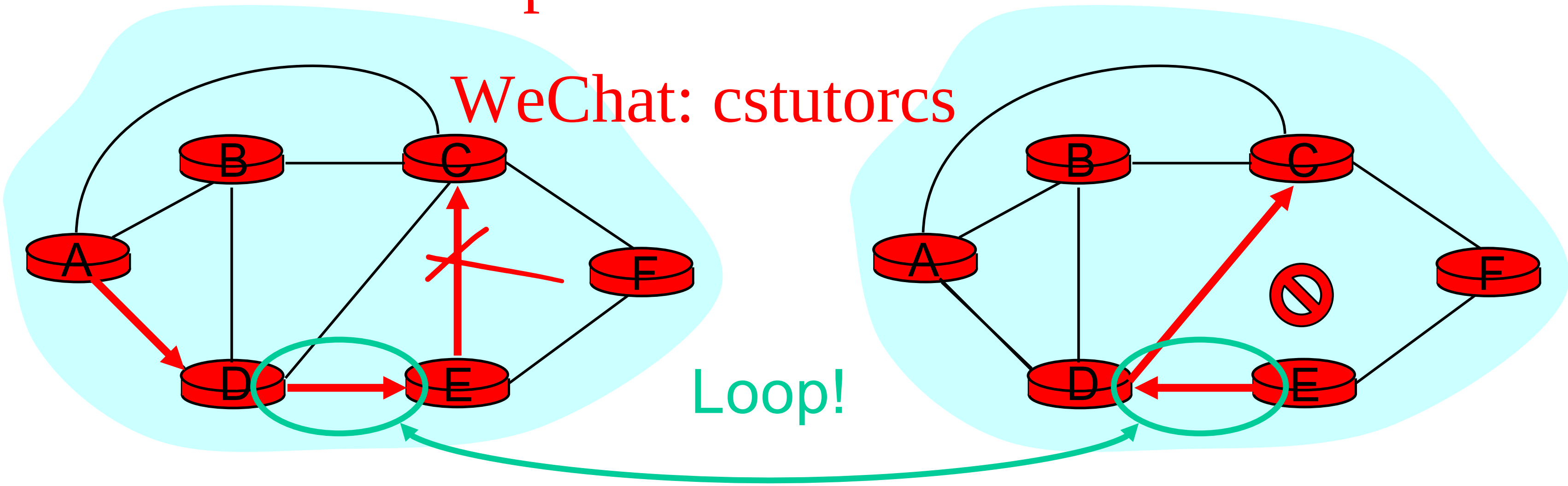
- How many entries in the LS topology database? O(E)

- How many entries in the forwarding table? O(N)

Issue#2: Transient Disruptions

- Inconsistent link-state database
  - Some routers know about failure before others
  - The shortest paths are no longer consistent
  - Can cause **transient** forwarding loops

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



A and D think that this is the path to C

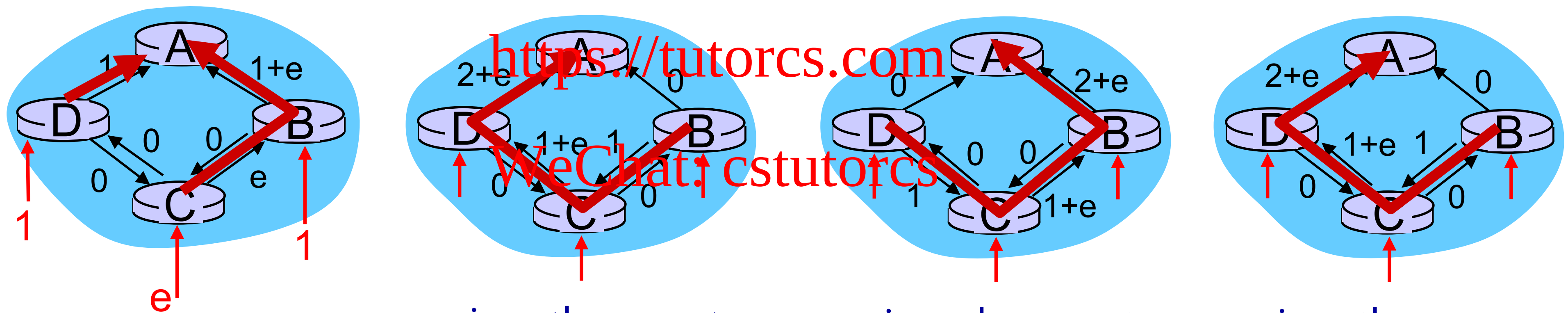
E thinks that this is the path to C

Oscillations

oscillations possible:

- e.g., suppose link cost equals amount of carried traffic:

Assignment Project Exam Help



https://tutorcs.com

WeChat: cstutorcs

initially

given these costs, find new routing…. resulting in new costs

given these costs, find new routing…. resulting in new costs

given these costs, find new routing…. resulting in new costs

Network layer, control plane: outline

5.1 introduction

5.2 routing protocols

- link state
- distance vector
- hierarchical routing

5.6 ICMP: The Internet Control Message Protocol

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Distance vector algorithm

Bellman-Ford equation

Assignment Project Exam Help

let

$d_x(y) :=$ cost of least-cost path from x to y

https://tutorcs.com

then

WeChat: cstutorcs

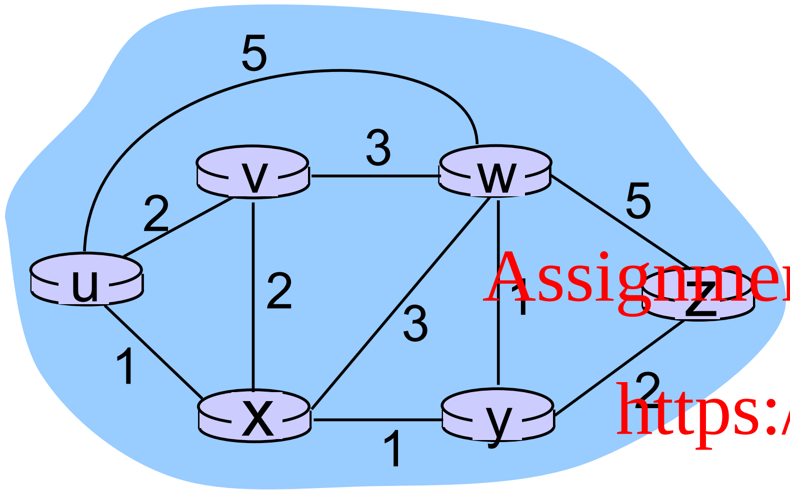
$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

cost from neighbor v to destination y

cost to neighbor v

min taken over all neighbors v of x

Bellman-Ford example



clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

Assignment Project Exam Help

B-F equation says:

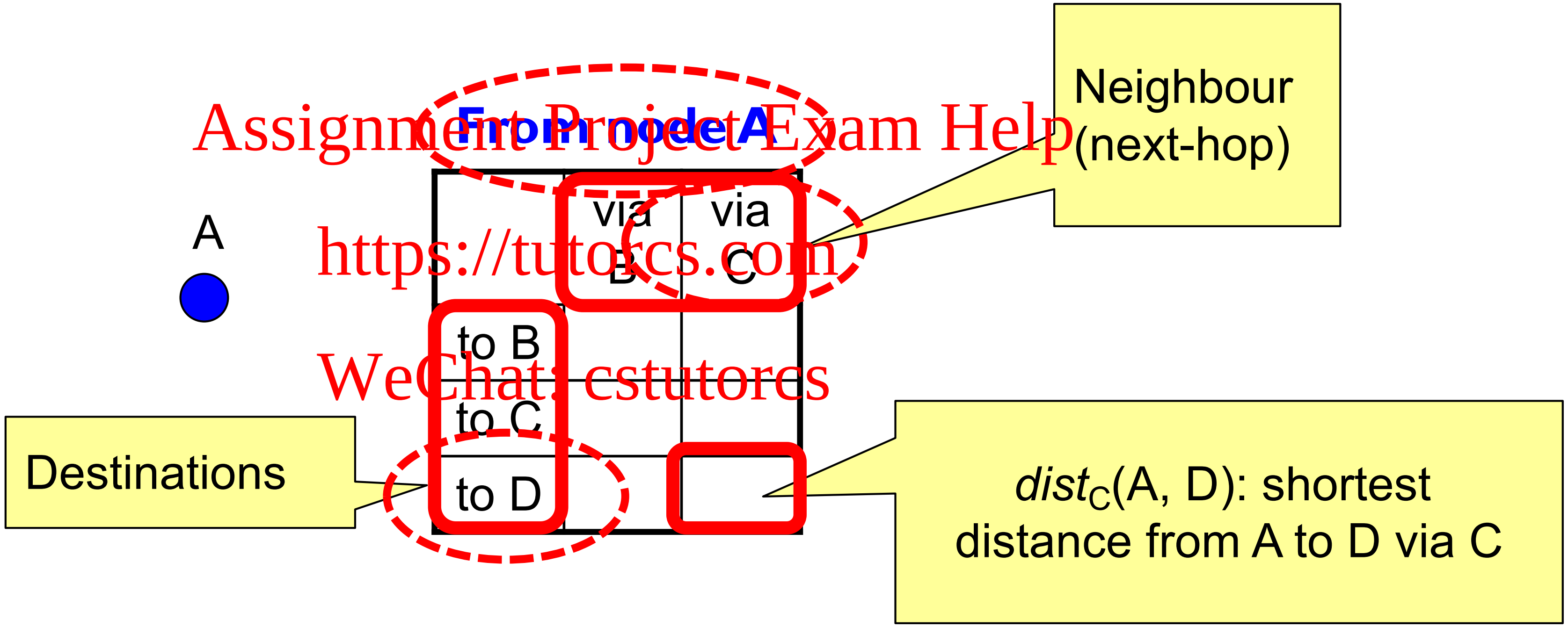
https://tutorcs.com

WeChat: cstutorcs

$$
\begin{aligned}
d_u(z) &= \min \{ c(u,v) + d_v(z), \\
&\qquad c(u,x) + d_x(z), \\
&\qquad c(u,w) + d_w(z) \} \\
&= \min \{2 + 5, \\
&\qquad 1 + 3, \\
&\qquad 5 + 3\} = 4
\end{aligned}
$$

node achieving minimum is next hop in shortest path, used in forwarding table

How Distance-Vector (DV) works



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Each router maintains its shortest distance to every destination via each of its neighbours

How Distance-Vector (DV) works

MIN { $dist_B(A,B)$, $dist_C(A, B)$ }

A

From node A

	via B	via C
to B		
to C		
to D		

A's distance vector (DV)

	min dist
to B	
to C	
to D	

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Each router computes its shortest distance to every destination via any of its neighbors

How Distance-Vector (DV) works

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

A

From node A

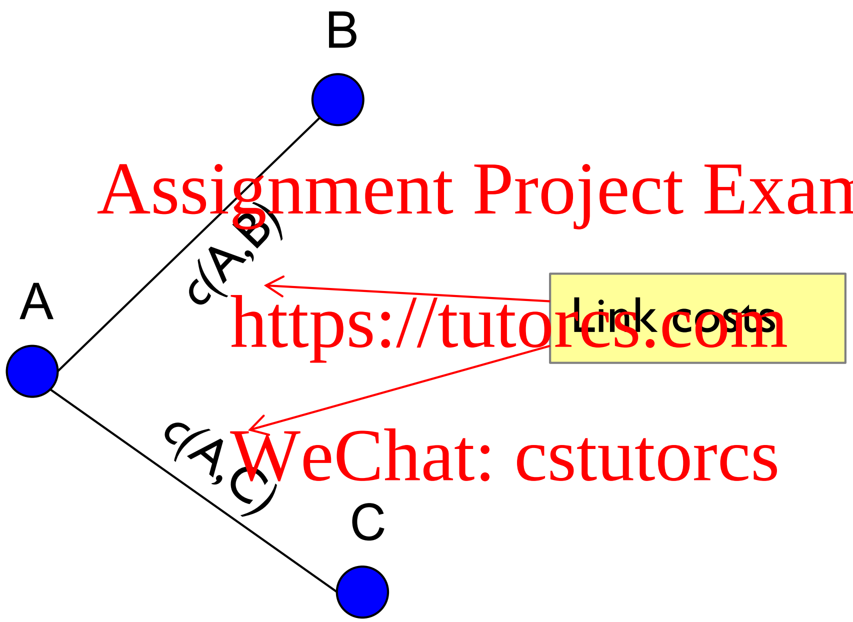
	via B	via C
to B	?	?
to C	?	?
to D	?	?

A's DV

	min dist
to B	?
to C	?
to D	?

How does A initialize its dist() table and DV?

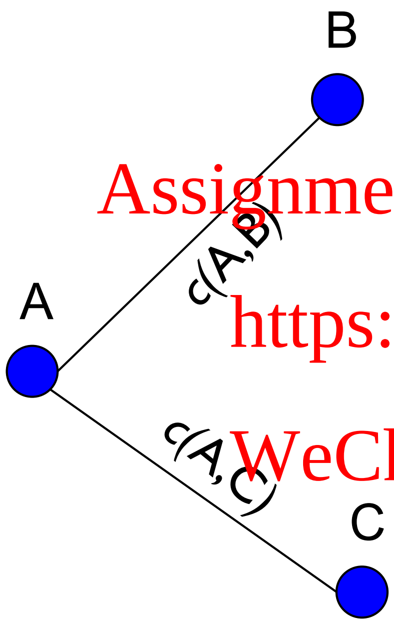
How Distance-Vector (DV) works



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

How does A initialize its dist() table and DV?

How Distance-Vector (DV) works



Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

From node A

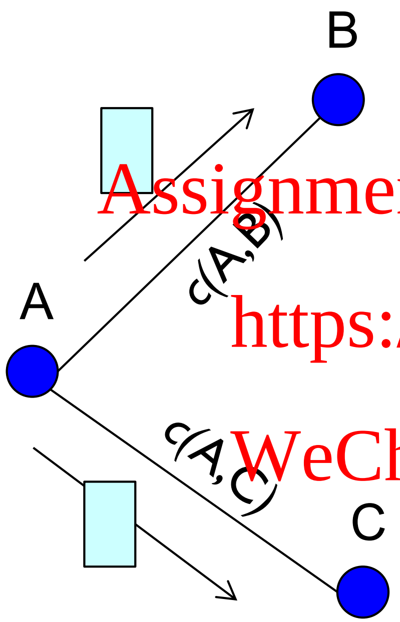
	via B	via C
to B	*c*(A,B)	∞
to C	∞	*c*(A,C)
to D	∞	∞

A's DV

	mindist
to B	c(A,B)
to C	c(A,C)
to D	∞

Each router initializes its *dist*() table based on its immediate neighbors and link costs

How Distance-Vector (DV) works



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Assume that A's DV is as follows at some later time

From node A

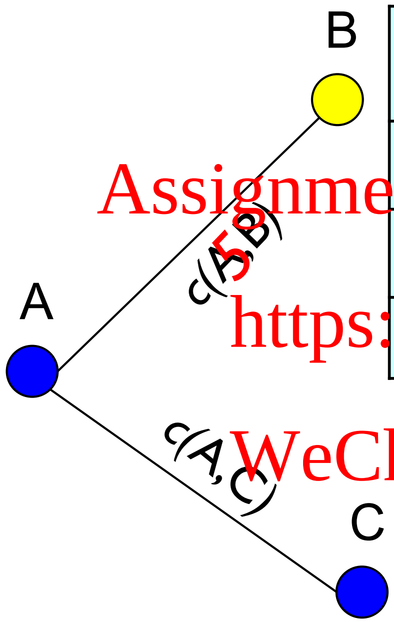
	via B	via C
to B	c(A,B)	∞
to C	∞	c(A,C)
to D	∞	∞

A's DV

	mindist
to B	5
to C	6
to D	2

Each router sends its DV to its immediate neighbors

How Distance-Vector (DV) works



A's DV

	min
to B	5
to C	6
to D	2

From node B

	via A	via C
to A	5	∞
to C	15	1
to D	∞	∞

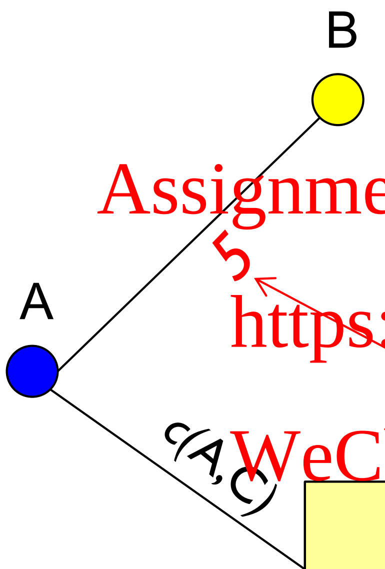
B's DV

	mindist
to A	5
to C	1
to D	∞

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Routers process received DVs

How Distance-Vector (DV) works



A's DV

	min
to B	5
to C	6
to D	2

From node B

	via A	via C
to A	5	∞
to C	11	1
to D	7	∞

B's DV

	mindist
to A	5
to C	1
to D	7

new = c(B,A) + *mindist*(A, D)

Routers process received DVs

And repeat…

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Distance Vector Routing

- Each router knows the links to its neighbors
- Each router has provisional "shortest path" to every other router -- its distance vector (DV)
- Routers exchange this DV with their neighbors
- Routers look over the set of options offered by their neighbors and select the best one
- Iterative process converges to set of shortest paths

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Distance vector routing

iterative, asynchronous:
each local iteration caused by:

- local link cost change
- DV update message from neighbor

distributed:

- each node notifies neighbors *only* when its DV changes
  - neighbors then notify their neighbors if necessary

each node:

wait for (change in local link cost or msg from neighbor)

↓

recompute estimates

↓

if DV to any dest has changed, *notify* neighbors

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Distance Vector

- $c(i,j)$: link cost from node *i* to *j*
- $dist_Z(A,V)$: shortest dist. from A to V via Z
- $mindist(A,V)$: shortest dist. from A to V

At node A

```
Initialization:
    for all destinations V do
        if V is neighbor of A
            dist_V(A, V) = mindist(A,V) = c(A,V);
        else
            dist_V(A, V) = mindist(A,V) = ∞;
    send mindist(A, *) to all neighbors
loop:
    wait (until A sees a link cost change to neighbor V  /* case 1 */
        or until A receives mindist(V,*) from neighbor V)   /* case 2 */
    if (c(A,V) changes by ±d)  /* ⇐ case 1 */
        for all destinations Y do
            dist_V(A,Y) = dist_V(A,Y) ± d
    else /* ⇐ case 2: */
        for all destinations Y do
            dist_V(A,Y) = c(A,V) + mindist(V, Y);
    update mindist(A,*)
    if (there is a change in mindist(A, *))
        send mindist(A, *) to all neighbors
forever
```

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Distance Vector

- c(i,j): link cost from node *i* to *j*
- $dist_Z(A,V)$: shortest dist. from A to V via Z
- mindist(A,V): shortest dist. from A to V

```
At node A
Initialization:
  for all destinations V do
    if V is neighbor of A
      dist_V(A, V) = mindist(A,V) = c(A,V);
    else
      dist_V(A, V) = mindist(A,V) = ∞;
  send mindist(A, *) to all neighbors
loop:
  wait (until A sees a link cost change to neighbor V  /* case 1 */
       or until A receives mindist(V,*) from neighbor V)   /* case 2 */
  if (c(A,V) changes by ±d)  /* ⇐ case 1 */
    for all destinations Y do
        dist_V(A,Y) =  dist_V(A,Y) ± d
  else /* ⇐ case 2: */
    for all destinations Y do
        dist_V(A,Y) = c(A,V) + mindist(V, Y);
  update mindist(A,*)
  if (there is a change in mindist(A, *))
       send mindist(A, *) to all neighbors
forever
```

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Example: Initialization

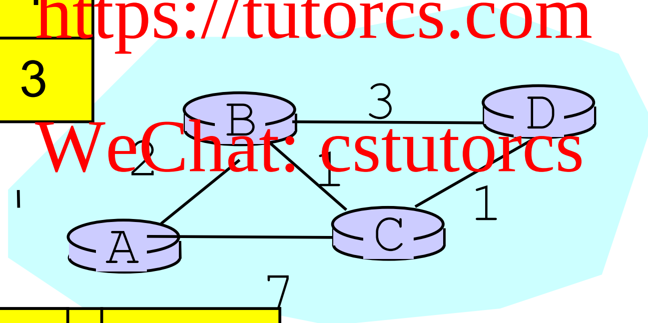
from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



from Node A

	via B	via C	min dist	min dist
to A	-	-	0	0
to B	2	∞	2	2
to C	∞	7	7	7
to D	∞	∞	∞	∞

from Node C

	via A	via B	via D	min dist
to A	7	∞	∞	7
to B	∞	1	∞	1
to C	-	-	-	0
to D	∞	∞	1	1

Example: C sends update to A

from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

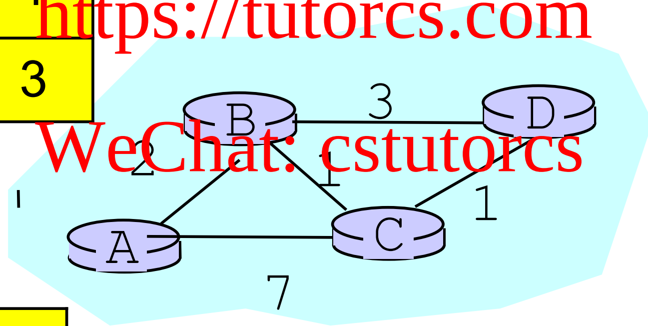
from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	∞	2
to C	∞	7	7
to D	∞	∞	∞

from Node C

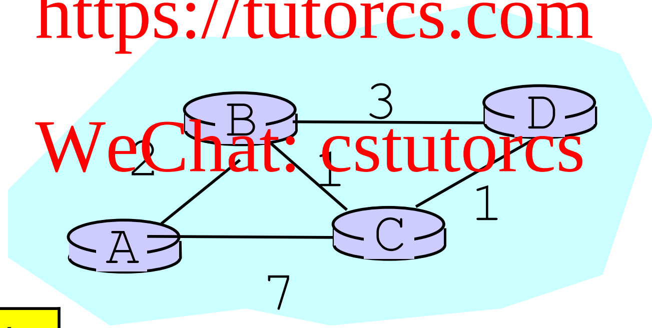
	via A	via B	via D	min dist
to A	7	∞	∞	7
to B	∞	1	∞	1
to C	-	-	-	0
to D	∞	∞	1	1

Example: C sends update to A

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C
to A	-	-
to B	2	∞
to C	∞	7
to D	∞	∞

min dist
0
2
7
∞

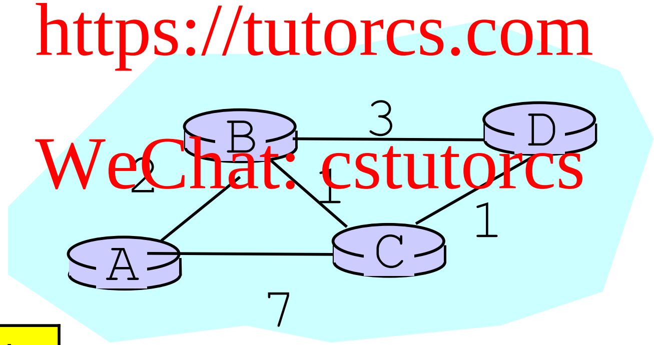
min dist
7
1
0
1

Example: C sends update to A

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C
to A	-	-
to B	2	**8**
to C	∞	7
to D	∞	**8**

min dist
0
2
7
∞

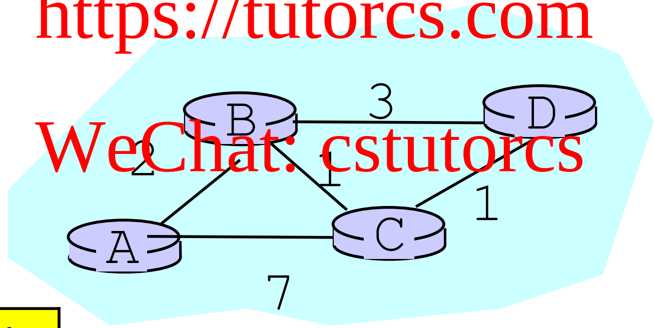
min dist
7
1
0
1

Example: C sends update to A

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C
to A	-	-
to B	2	**8**
to C	∞	7
to D	∞	**8**

min dist
0
2
7
8

Example: now B sends update to A

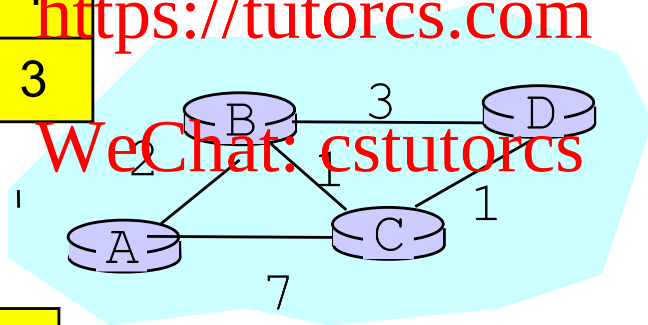
from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	∞	7	7
to D	∞	8	8

from Node C

	via A	via B	via D	min dist
to A	7	∞	∞	7
to B	∞	1	∞	1
to C	-	-	-	0
to D	∞	∞	1	1

Example: now B sends update to A

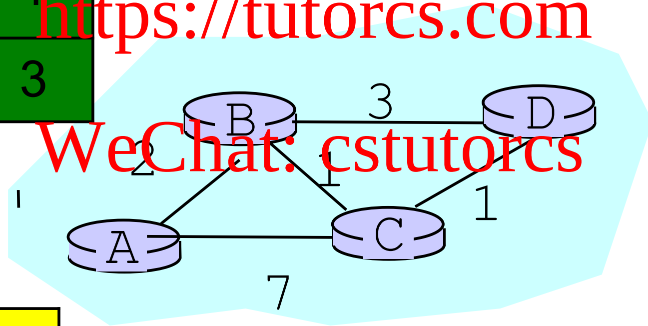
from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	∞	7	7
to D	∞	8	8

from Node C

	via A	via B	via D	min dist
to A	7	∞	∞	7
to B	∞	1	∞	1
to C	-	-	-	0
to D	∞	∞	1	1

Example: now B sends update to A

from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

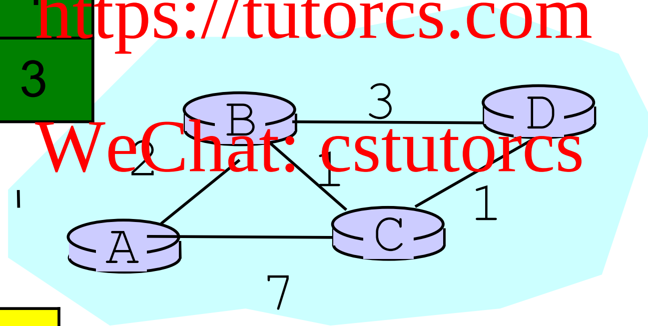
from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	7
to D	5	8	8

from Node C

	via	via	via	min dist
	[illegible]	[illegible]	[illegible]	7
	[illegible]	[illegible]	[illegible]	1
to C	-	-	-	0
to D	∞	∞	1	1

Make sure you know why this is 5, not 4!

Example: now B sends update to A

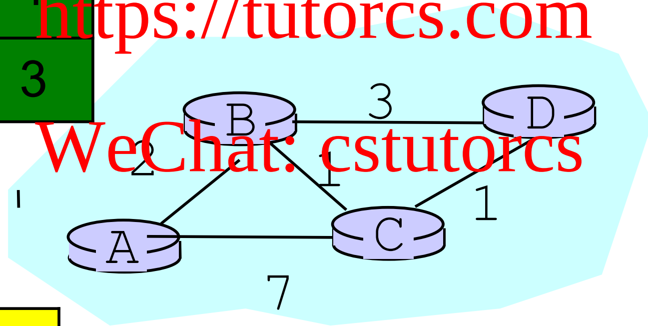
from Node B

	via A	via C	via D	min dist
to A	2	∞	∞	2
to B	-	-	-	0
to C	∞	1	∞	1
to D	∞	∞	3	3

from Node D

	via B	via C	min dist
to A	∞	∞	∞
to B	3	∞	3
to C	∞	1	1
to D	-	-	0

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	**3**	7	**3**
to D	**5**	8	**5**

from Node C

	via A	via B	via D	min dist
to A	7	∞	∞	7
to B	∞	1	∞	1
to C	-	-	-	0
to D	∞	∞	1	1

All nodes know the best two-hop paths.

Make sure you believe this

from Node B

	via A	via C	via D	min dist
to A	2	8	∞	2
to B	-	-	-	0
to C	9	1	4	1
to D	∞	2	3	**2**

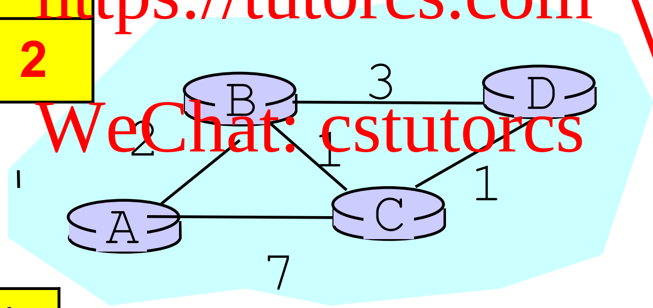
from Node D

	via B	via C	min dist
to A	5	8	**5**
to B	3	2	**2**
to C	4	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	**3**
to D	5	8	**5**

from Node C

	via A	via B	via D	min dist
to A	7	3	∞	**3**
to B	9	1	4	1
to C	-	-	-	0
to D	∞	4	1	1

Example: Now A sends update to B

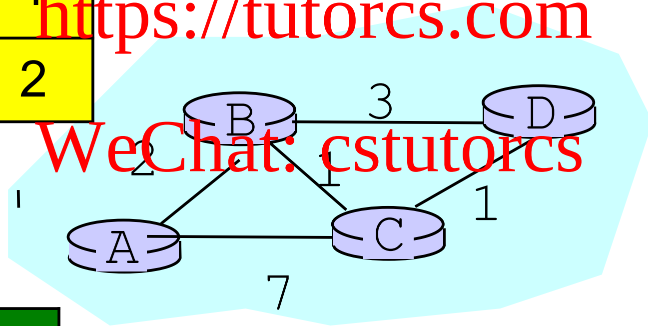
from Node B

	via A	via C	via D	min dist
to A	2	8	∞	2
to B	-	-	-	0
to C	9	1	4	1
to D	∞	2	3	2

from Node D

	via B	via C	min dist
to A	5	8	5
to B	3	2	2
to C	4	1	1
to D	-	-	0

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	3
to D	5	8	5

from Node C

	via A	via B	via D	min dist
to A	7	3	∞	3
to B	9	1	4	1
to C	-	-	-	0
to D	∞	4	1	1

Example: Now

Updated

from Node B

	via A	via C	via D	min dist
to A	2	8	∞	[illegible]
to B	-	-	[illegible]	0
to C	5	1	4	1
to D	7	2	3	2

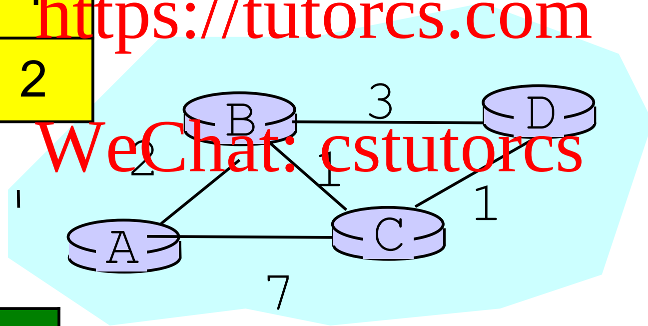
from Node D

	via B	via C	min dist
to A	5	8	5
to B	3	2	2
to C	4	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	3
to D	5	8	5

from Node C

	via A	via B	via D	min dist
to A	7	3	∞	3
to B	9	1	4	1
to C	-	-	-	0
to D	∞	4	1	1

Check: *All nodes know the best three-hop paths.*

from Node B

	via A	via C	via D	min dist
to A	2	4	8	2
to B	-	-	-	0
to C	5	1	4	1
to D	7	2	3	2

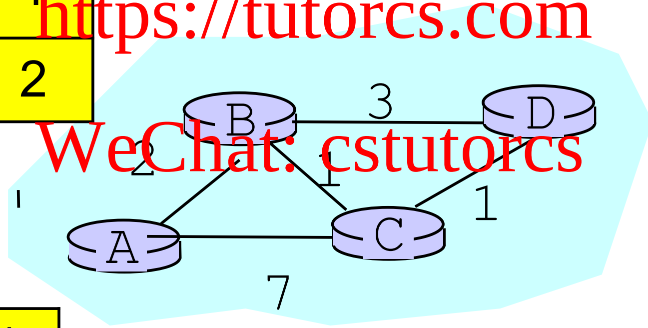
from Node D

	via B	via C	min dist
to A	5	4	4
to B	3	2	2
to C	4	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	3
to D	4	8	4

Check

from Node C

	via A	via B	via D	min dist
to A	7	3	6	3
to B	9	1	3	1
to C	-	-	-	0
to D	12	3	1	1

Example: End of 3rd Full Exchange

No further change in DVs → Convergence!

from Node B

	via A	via C	via D	min dist
to A	2	4	7	2
to B	-	-	-	0
to C	5	1	4	1
to D	6	2	3	2

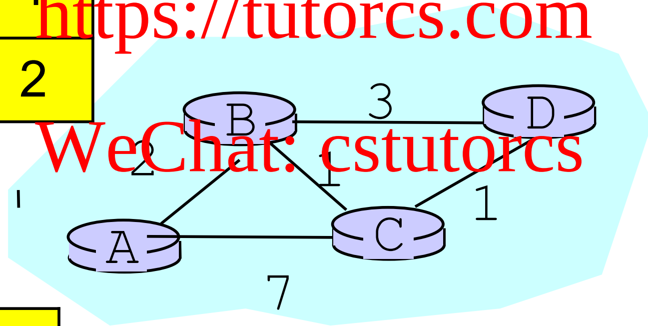
from Node D

	via B	via C	min dist
to A	5	4	4
to B	3	2	2
to C	4	1	1
to D	-	-	0

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



from Node A

	via B	via C	min dist
to A	-	-	0
to B	2	8	2
to C	3	7	3
to D	4	8	4

from Node C

	via A	via B	via D	min dist
to A	7	3	5	3
to B	9	1	3	1
to C	-	-	-	0
to D	11	3	1	1

Intuition

- Initial state: best one-hop paths
- One simultaneous round: best two-hop paths
- Two simultaneous rounds: best three-hop paths
- …
- Kth simultaneous round: best (k+1) hop paths

- Must eventually converge
  - as soon as it reaches longest best path
- …..but how does it respond to changes in cost?

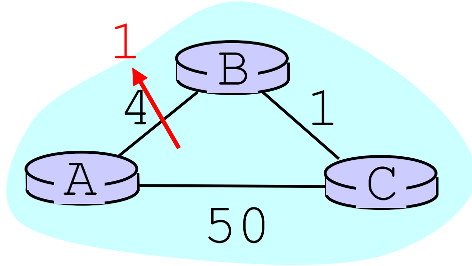
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Problems with Distance Vector

- A number of problems can occur in a network using distance vector algorithm
- Most of these problems are caused by slow convergence or routers converging on incorrect information
- *Convergence* is the time during which all routers come to an agreement about the best paths through the internetwork
  - whenever topology changes there is a period of instability in the network as the routers converge
- Reacts rapidly to good news, but leisurely to bad news

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

DV: Link Cost Changes



via / to

Stable state

Node A	B	C
B	4	51
C	5	50

Node B	A	C
A	4	6
C	9	1

Node C	A	B
A	50	5
B	54	1

A-B changed

Node A	B	C
B	1	51
C	2	[illegible]

Node B	A	C
A	1	6
C	6	1

Node C	A	B
A	50	5
B	54	1

A sends its DV to B, C

Node A	B	C
B	1	51
C	[illegible]	[illegible]

Node B	[illegible]	[illegible]
A	[illegible]	[illegible]
C	3	1

Node C	A	B
A	50	5
B	51	1

B sends its DV to A, C

Node A	B	C
B	1	51
C	[illegible]	[illegible]

Node B	[illegible]	[illegible]
A	[illegible]	[illegible]
C	3	1

Node C	A	B
A	50	2
B	51	1

C sends its DV to A, B

Node A	B	C
B	1	51
C	[illegible]	50

Node B	[illegible]	C
A	[illegible]	3
C	3	1

Node C	A	B
A	50	2
B	51	1

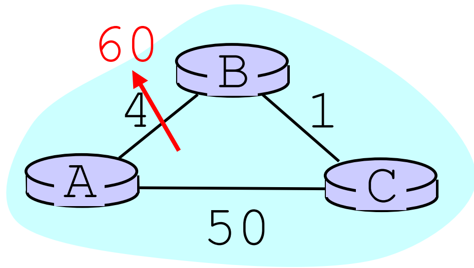
deduct 3 from distances $dist_B(A,*)$ and $dist_A(B,*)$

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Link cost changes here

"good news travels fast"

DV: Link Cost Changes



Stable state (via / to)

Node A

	B	C
B	4	51
C	5	50

Node B

	A	C
A	4	6
C	9	1

Node C

	A	B
A	50	5
B	54	1

A-B changed

Node A

	B	C
B	60	51
C	61	50

Node B

	A	C
A	60	6
C	65	1

Node C

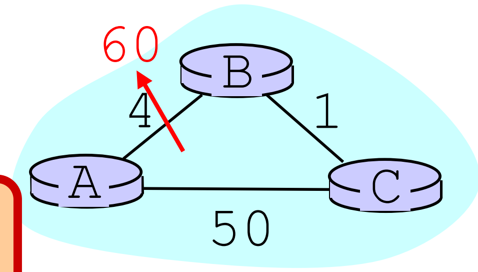
	A	B
A	50	5
B	54	1

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

add 56 to distances $dist_B(A,*)$ and $dist_A(B,*)$

Link cost changes here

DV: Link Cost Changes



This is the "Counting to Infinity" Problem

Node A (to: rows; via: columns)

Stable state	B	C
B	4	51
C	5	50

A-B changed	B	C
B	60	51
C	61	50

A sends its DV to B, C	B	C
B	60	51
C	61	50

B sends its DV to A, C	B	C
B	60	51
C	61	50

C sends its DV to A, B	B	C
B	60	51
C	61	50

Node B

Stable state	A	C
A	4	6
C	9	1

A-B changed	A	C
A	60	6
C	65	1

A sends its DV to B, C	A	C
A	60	6
C	110	1

B sends its DV to A, C	A	C
A	60	6
C	110	1

C sends its DV to A, B	A	C
A	60	8
C	110	1

Node C

Stable state	A	B
A	50	5
B	54	1

A-B changed	A	B
A	50	5
B	54	1

A sends its DV to B, C	A	B
A	50	5
B	101	1

B sends its DV to A, C	A	B
A	50	7
B	101	1

C sends its DV to A, B	A	B
A	50	7
B	101	1

Link cost changes here

"bad news travels slowly"
(not yet converged)

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

The "Poisoned Reverse" Rule

- Heuristic to avoid count-to-infinity

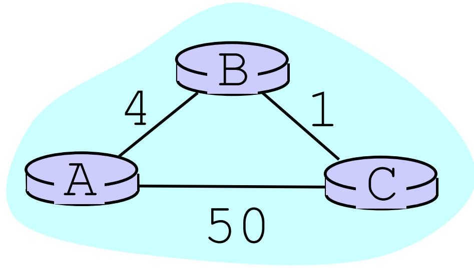
- If B routes via C to get to A:
  - B tells C its (B's) distance to A is infinite (so C won't route to A via B)

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

DV: Poisoned Reverse

If B routes through C to get to A:

B tells C its (B's) distance to A is infinite



Stable state

via

Node A (to)

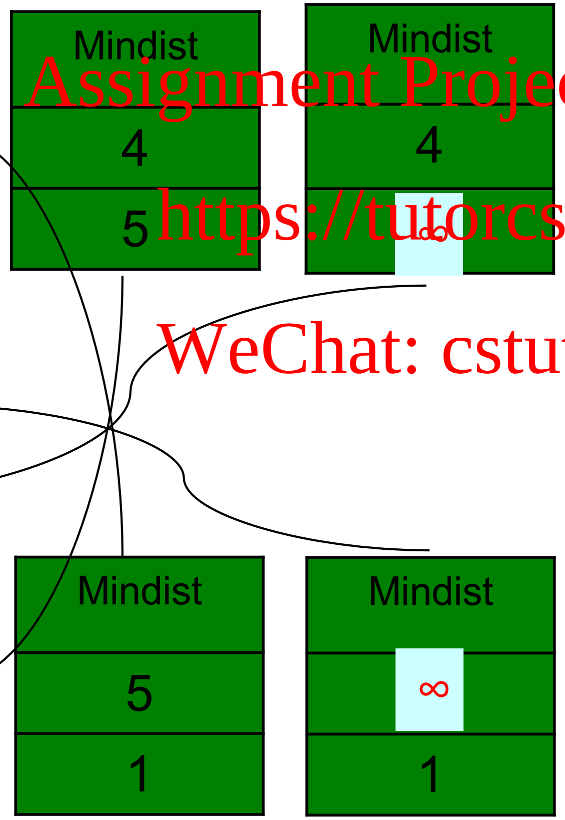
	B	C
B	4	51
C	5	50

Node B

	A	C
A	4	∞
C	∞	1

Node C

	A	B
A	50	5
B	54	1

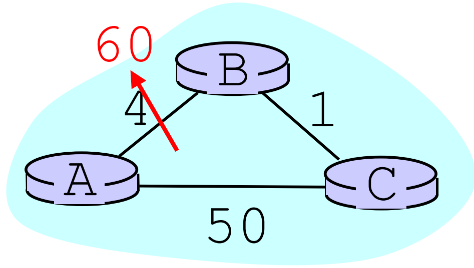


Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

DV: Poisoned Reverse

If B routes through C to get to A:

B tells C its (B's) distance to A is infinite



Stable state | A-B changed

via

to

Node A

Stable state:

	B	C
B	4	51
C	5	50

A-B changed:

	B	C
B	60	51
C	61	50

Node B

Stable state:

	A	C
A	4	∞
C	∞	1

A-B changed:

	A	C
A	60	6
C	65	1

Node C

Stable state:

	A	B
A	50	5
B	54	1

A-B changed:

	A	B
A	50	5
B	54	1

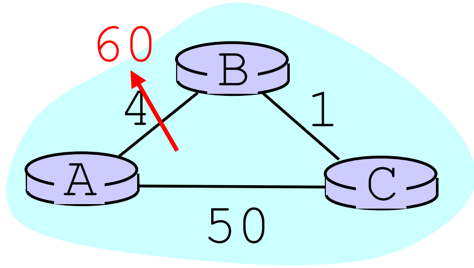
Link cost changes here

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

DV: Poisoned Reverse

If B routes through C to get to A:

B tells C its (B's) distance to A is infinite



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

(rows: to; columns: via)

Node A

Stable state

	B	C
B	4	51
C	5	50

A-B changed

	B	C
B	60	51
C	61	50

A sends its DV to B, C

	B	C
B	60	51
C	61	50

B sends its DV to A, C

	B	C
B	60	51
C	61	50

Node B

Stable state

	A	C
A	4	∞
C	∞	1

A-B changed

	A	C
A	60	∞
C	∞	1

A sends its DV to B, C

	A	C
A	60	∞
C	110	1

B sends its DV to A, C

	A	C
A	60	∞
C	110	1

Node C

Stable state

	A	B
A	50	5
B	54	1

A-B changed

	A	B
A	50	5
B	54	1

A sends its DV to B, C

	A	B
A	50	5
B	∞	1

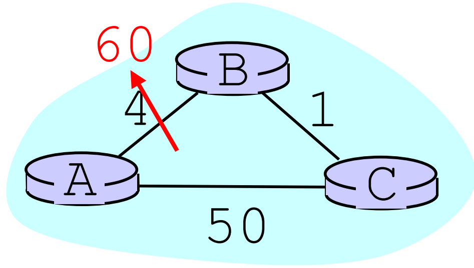
B sends its DV to A, C

	A	B
A	50	7
B	∞	1

Link cost changes here

DV: Poisoned Reverse

If B routes through C to get to A:
B tells C its (B's) distance to A is infinite



Columns are "via", rows are "to".

Node A

Stable state	B	C		A-B changed	B	C		A sends its DV to B, C	B	C		B sends its DV to A, C	B	C
B	4	51		B	60	51		B	60	51		B	60	51
C	5	50		C	61	50		C	61	50		C	61	50

Node B

	A	C			A	C			A	C			A	C
A	4	∞		A	60	∞		A	60	∞		A	60	∞
C	∞	1		C	∞	1		C	110	1		C	110	1

Node C

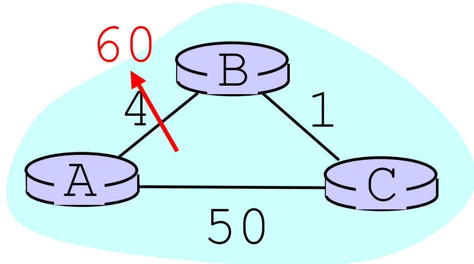
	A	B			A	B			A	B			A	B
A	50	5		A	50	5		A	50	5		A	50	61
B	54	1		B	54	1		B	∞	1		B	∞	1

Link cost changes here

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

DV: Poisoned Reverse

If B routes through C to get to A:
B tells C its (B's) distance to A is infinite



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Stable state

via

to

Node A

	B	C
B	4	51
C	5	50

Node B

	A	C
A	4	∞
C	∞	1

Node C

	A	B
A	50	5
B	54	1

A-B changed

Node A

	B	C
B	60	51
C	61	50

Node B

	A	C
A	60	∞
C	∞	1

Node C

	A	B
A	50	5
B	54	1

A sends its DV to B, C

Node A

	B	C
B	60	51
C	61	50

Node B

	A	C
A	60	∞
C	110	1

Node C

	A	B
A	50	5
B	∞	1

B sends its DV to A, C

Node A

	B	C
B	60	51
C	61	50

Node B

	A	C
A	60	∞
C	110	1

Node C

	A	B
A	50	61
B	∞	1

C sends its DV to A, B

Node A

	B	C
B	60	51
C	61	50

Node B

	A	C
A	60	51
C	110	1

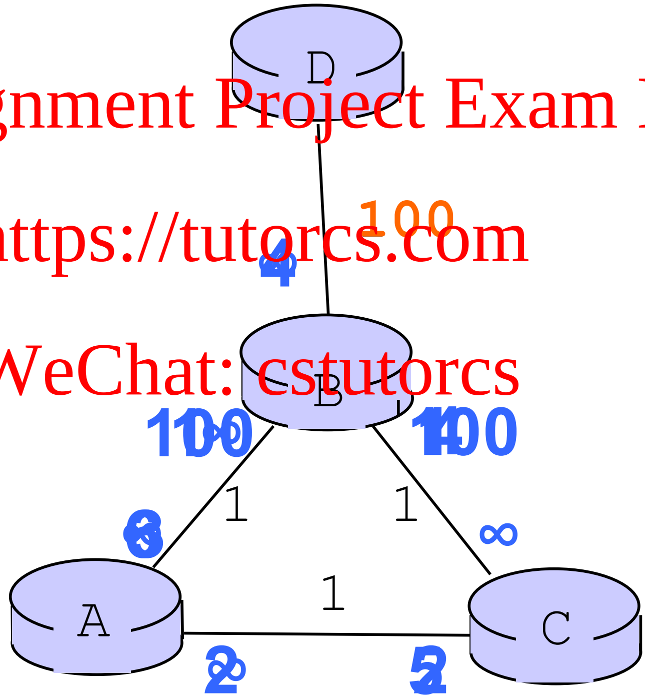
Node C

	A	B
A	50	∞
B	∞	1

Link cost changes here

Converges after C receives another update from B

Will Poison-Reverse Completely Solve the Count-to-Infinity Problem?



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Numbers in blue denote the best cost
to destination D advertised along the link

Quiz: Link-state routing

- In link state routing, each node sends information of its direct links (i.e., link state) to __________?

Assignment Project Exam Help

A. Immediate neighbours
B. All nodes in the network
https://tutorcs.com
C. Any one neighbor
WeChat: cstutorcs
D. No one

Answer: B

www.zeetings.com/salil

Quiz: Distance-vector routing

- In distance vector routing, each node shares its distance table with __________?

Assignment Project Exam Help

A. All Immediate neighbours
B. All nodes in the network
https://tutorcs.com
C. Any one neighbor
WeChat: cstutorcs
D. No one

Answer: A

www.zeetings.com/salil

Quiz: Distance-vector routing

- Which of the following is true of distance vector routing?

Assignment Project Exam Help

A. Convergence delay depends on the topology (nodes and links) and link weights

https://tutorcs.com

B. Convergence delay depends on the number of nodes and links

WeChat: cstutorcs

C. Each node knows the entire topology

D. A and C

E. B and C

Answer: A

www.zeetings.com/salil

Comparison of LS and DV algorithms

message complexity

- ***LS:*** with n nodes, E links, O(nE) msgs sent
- ***DV:*** exchange between neighbors only
  - convergence time varies

speed of convergence

- ***LS:*** $O(n^2)$ algorithm requires O(nE) msgs
  - may have oscillations
- ***DV:*** convergence time varies
  - may be routing loops
  - count-to-infinity problem

robustness: what happens if router malfunctions?

LS:

- node can advertise incorrect *link* cost
- each node computes only its *own* table

DV:

- DV node can advertise incorrect *path* cost
- each node's table used by others
  - error propagate thru network

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Real Protocols

Link State	*Distance Vector*
Open Shortest Path First (OSPF)	Routing Information Protocol (RIP)
Intermediate system to intermediate system (IS-IS)	Interior Gateway Routing Protocol (IGRP-Cisco)
	Border Gateway Protocol (BGP)

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Network layer, control plane: outline

5.1 introduction

5.2 routing protocols

- link state
- distance vector
- hierarchical routing

Self study (not on exam)

5.6 ICMP: The Internet Control Message Protocol

Assignment Project Exam Help

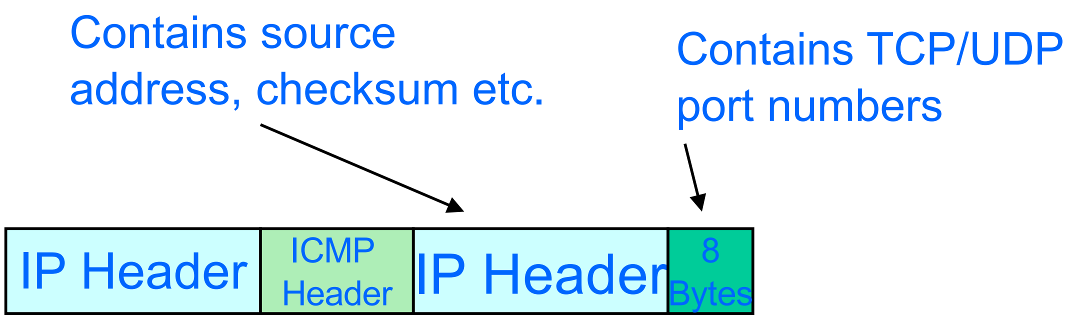
https://tutorcs.com

WeChat: cstutorcs

ICMP: Internet Control Message Protocol

- Used by hosts & routers to communicate network level infromation
  - Error reporting: unreachable host, network, port
  - Echo request/reply (used by ping)
- Works above IP layer
  - ICMP messages carried in IP datagrams
- ICMP message: type, code plus IP header and first 8 bytes of IP datagram payload causing error

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



ICMP: Internet Control Message Protocol

Type	Code	Description
0	0	echo reply(ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	3	dest port unreachable
3	4	frag needed; DF set
8	0	echo request(ping)
11	0	TTL expired
11	1	frag reassembly time exceeded
12	0	bad IP header

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

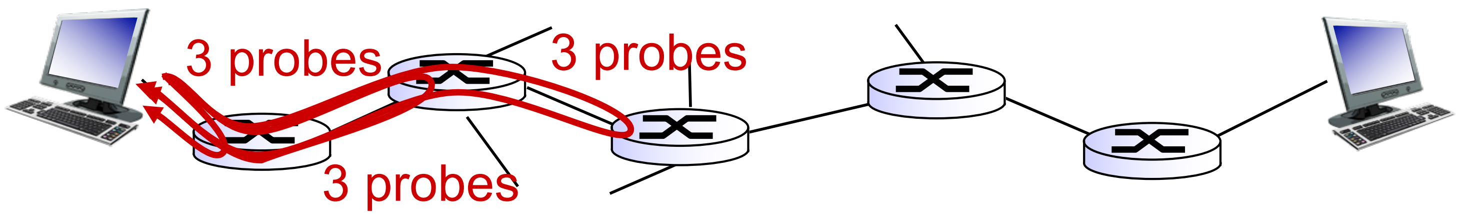
Traceroute and ICMP

- Source sends series of UDP segments to dest
  - first set has TTL =1
  - second set has TTL=2, etc.
  - unlikely port number
- When *n*th set of datagrams arrives to nth router:
  - router discards datagrams
  - and sends source ICMP messages (type 11, code 0)
  - ICMP messages includes IP address of router
- when ICMP messages arrives, source records RTTs

stopping criteria:

- UDP segment eventually arrives at destination host
- destination returns ICMP "port unreachable" message (type 3, code 3)
- source stops

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



Summary

- Network Layer: Data Plane
  - Overview
  - IP
- Network Layer: Control Plane
  - Routing Protocols
    - Link–state
    - Distance Vector
  - ICMP

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

COMP 3331/9331: Computer Networks and Applications

Assignment Project Exam Help

https://tutorcs.com

Week 9

WeChat: cstutorcs

Data link Layer

Reading Guide: Chapter 6, Sections 6.1 – 6.4, 6.7

Link layer and LANs

our goals:

- understand principles behind link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
  - local area networks: Ethernet

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Link layer, LANs: outline

6.1 introduction, services

6.2 error detection, correction

6.3 multiple access protocols

6.4 Switched LANs
- addressing, ARP
- Ethernet
- Switches
- VLANS (**EXCLUDED**)

6.5 link virtualization: MPLS (**EXCLUDED**)

6.6 data center networking (**EXCLUDED**)

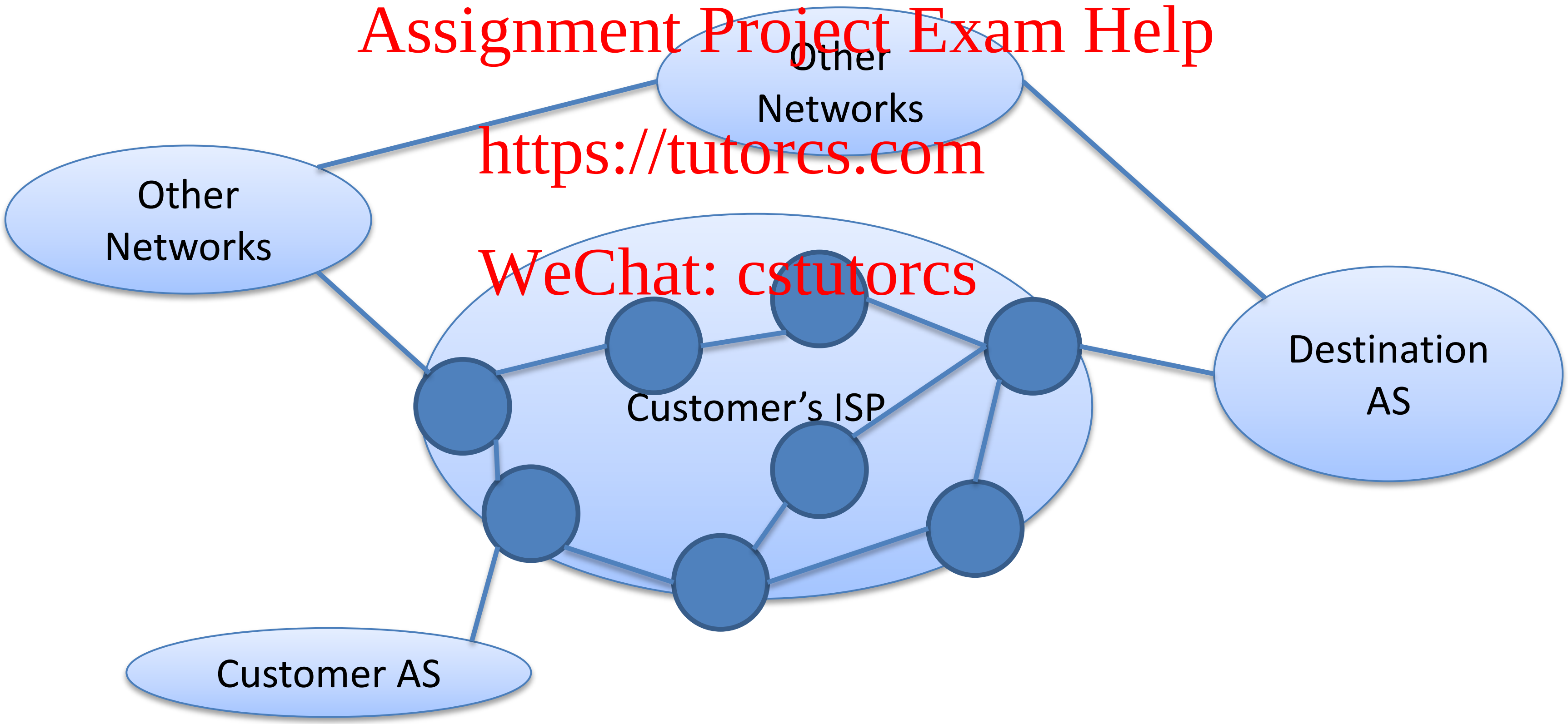
6.7 a day in the life of a web request

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

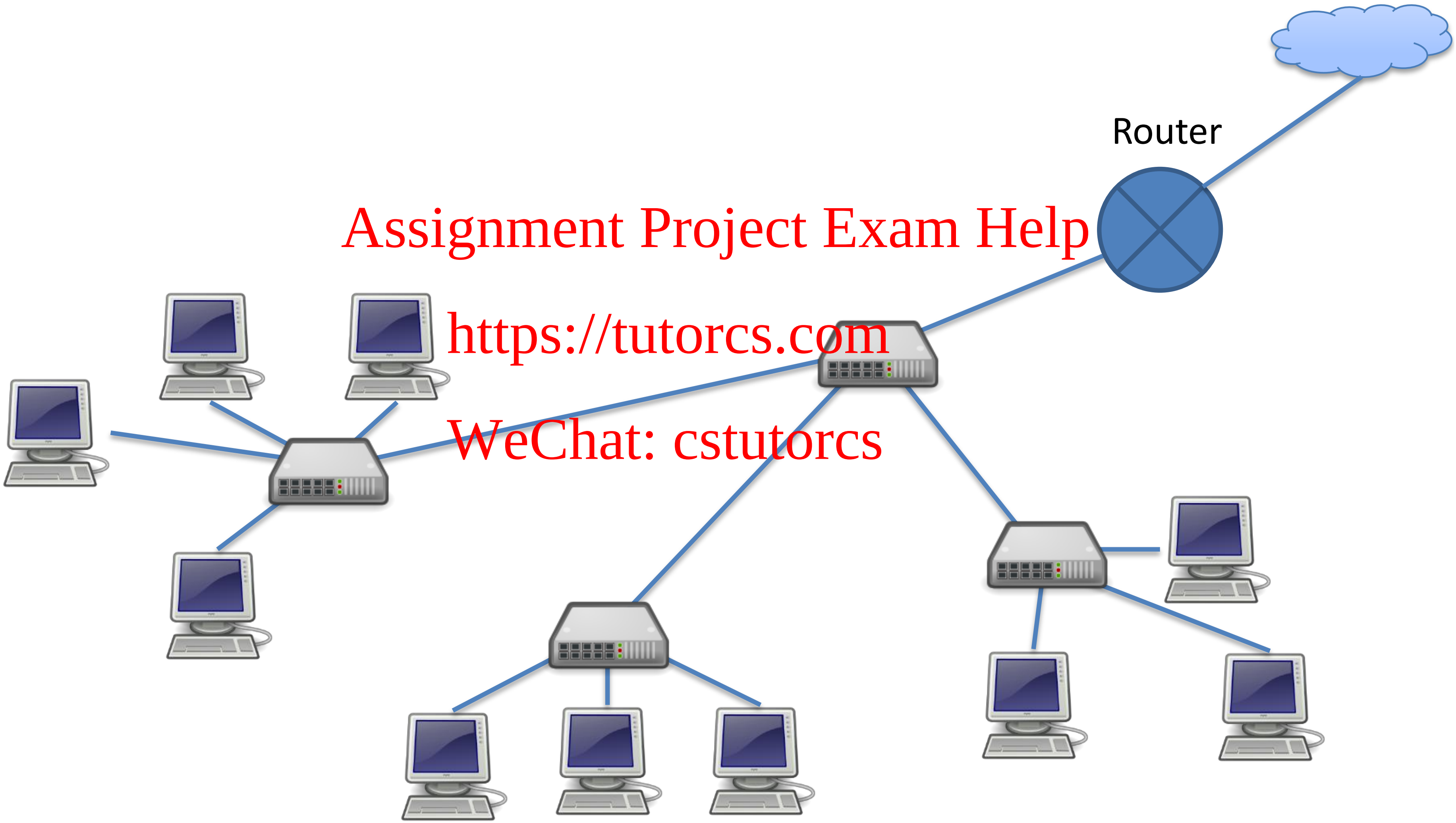
From Macro- to Micro-

- Previously, we looked at Internet scale…

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



Link layer focus: Within a Subnet



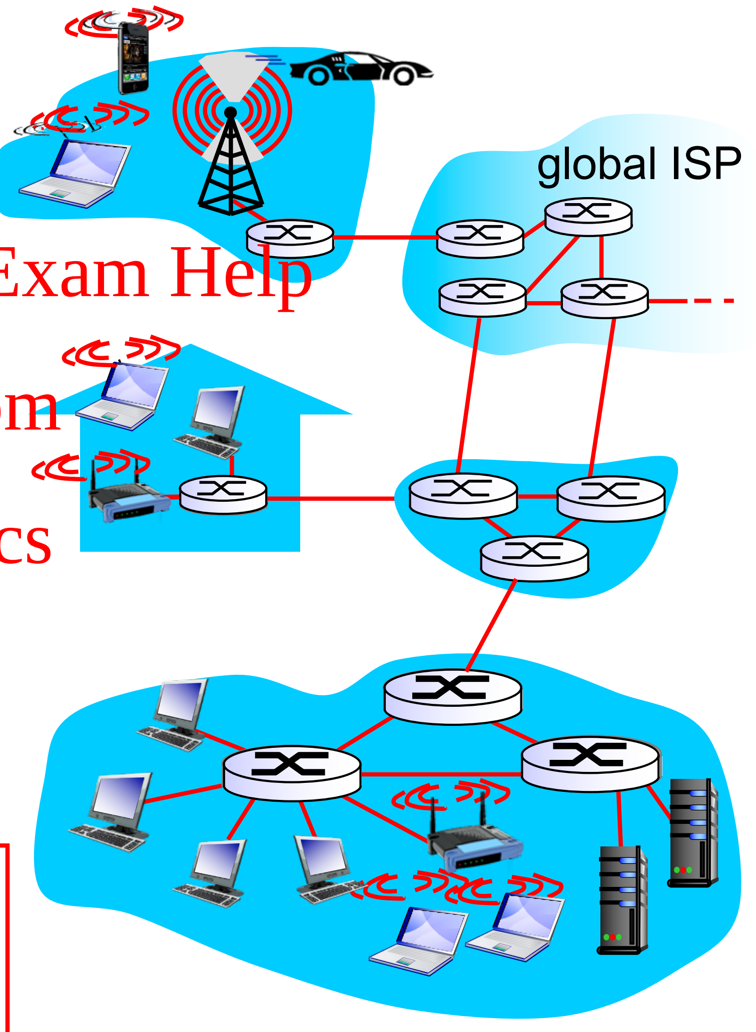
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Link layer: introduction

terminology:

- hosts and routers: nodes
- communication channels that connect adjacent nodes along communication path: links
  - wired links
  - wireless links
  - LANs
- layer-2 packet: *frame,* encapsulates datagram

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



data-link layer has responsibility of transferring datagram from one node to physically adjacent node over a link

Link layer: context

- datagram transferred by different link protocols over different links:
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- each link protocol provides different services
  - e.g., may or may not provide rdt over link

transportation analogy:

- trip from Princeton to Lausanne
  - limo: Princeton to JFK
  - plane: JFK to Geneva
  - train: Geneva to Lausanne
- tourist = datagram
- transport segment = communication link
- transportation mode = link layer protocol
- travel agent = routing algorithm

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Link layer services

- *framing, link access:*
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses used in frame headers to identify source, dest
    - different from IP address!
- *reliable delivery between adjacent nodes*
  - we learned how to do this already (chapter 3)!
  - seldom used on low bit-error link (fiber, some twisted pair)
  - wireless links: high error rates
    - *Q:* why both link-level and end-end reliability?

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Link layer services (more)

- *flow control:*
  - pacing between adjacent sending and receiving nodes
- *error detection:*
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- *error correction:*
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission
- *half-duplex and full-duplex*
  - with half duplex, nodes at both ends of link can transmit, but not at same time

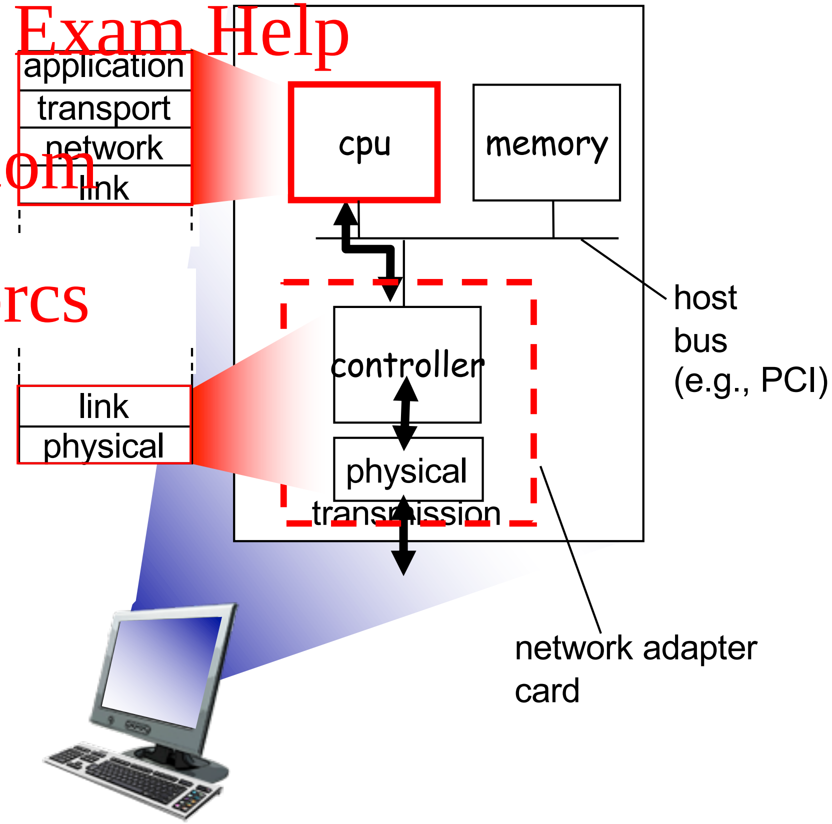
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Where is the link layer implemented?

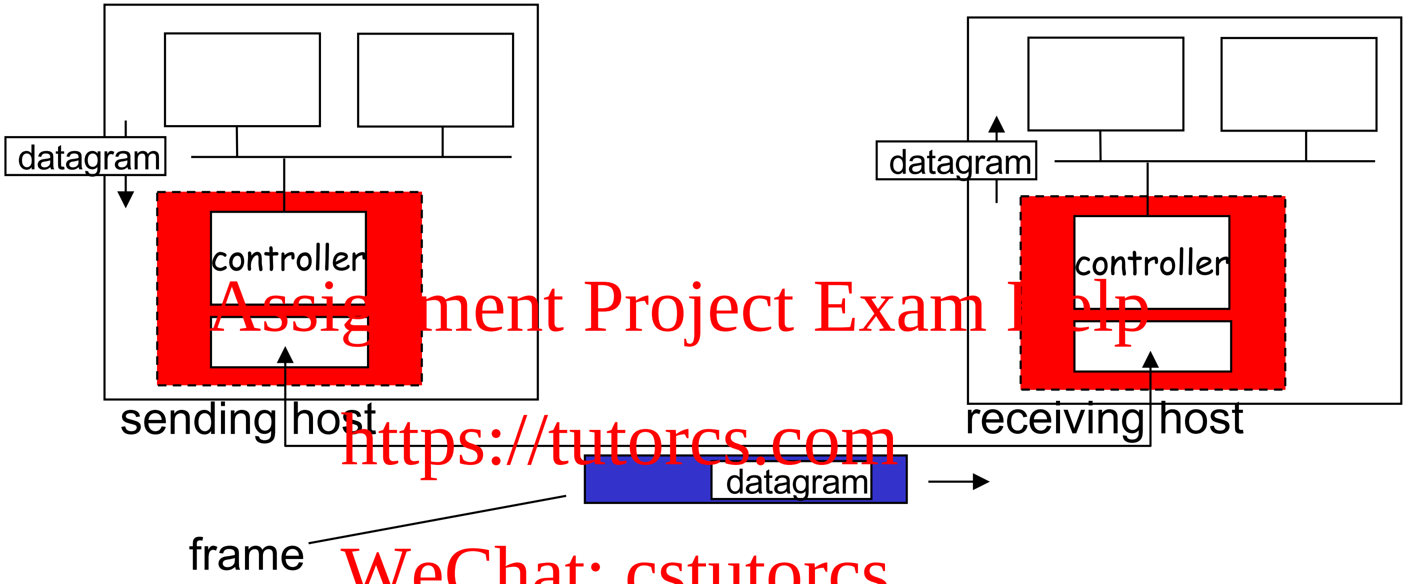
- in each and every host
- link layer implemented in "adaptor" (aka *network interface card* NIC) or on a chip
  - Ethernet card, 802.11 card; Ethernet chipset
  - implements link, physical layer
- attaches into host's system buses
- combination of hardware, software, firmware

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs





Adaptors communicating



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

- sending side:
  - encapsulates datagram in frame
  - adds error checking bits, rdt, flow control, etc.
- receiving side
  - looks for errors, rdt, flow control, etc
  - extracts datagram, passes to upper layer at receiving side

What is framing?

- Physical layer talks in terms of bits.
- How do we identify frames within the sequence of bits?
- Need to do Framing
  - Delimit the start and end of the frame
- Ethernet Framing
  - Timing/Physical layer

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Framing in Ethernet

- Start of frame is recognized by
  - Preamble : Seven bytes with pattern 10101010
  - Start of Frame Delimiter (SFD) : 10101011

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Preamble 7 Bytes	**SFD 1 Byte**	**Dest MAC 6 Bytes**	**Source MAC 6 Bytes**	**Type/Le ngth 2 Bytes**	**Payload 46-1500 Bytes**	**FCS/C RC 4 Bytes**	**Inter Frame Gap**

- Inter Frame Gap is 12 Bytes (96 bits) of idle state
  - 0.96 microsec for 100 Mbit/s Ethernet
  - 0.096 microsec for Gigabit/s Ethernet