CSCE 5580 Computer Networks
Assignment – 3

1. What is packet scheduling? Explain different types in detail with diagrams and why is it important in the network layer? (5 points)

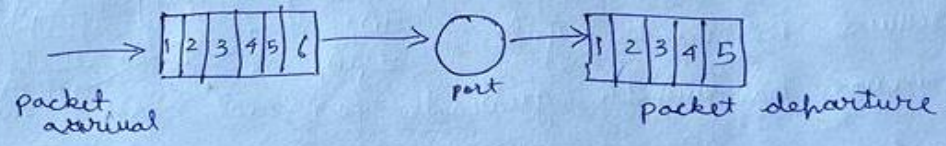
1 Packet Scheduling: Packet scheduling decides which packet to send next on a link. It is a process that determines the order in which the packets are transmitted over a network.

→ This happens in the Network layer.

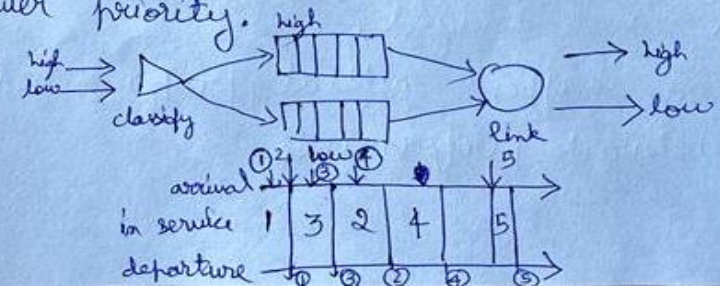
→ It ensures that packets are delivered efficiently with minimum delay and loss.

Types of Packet Scheduling

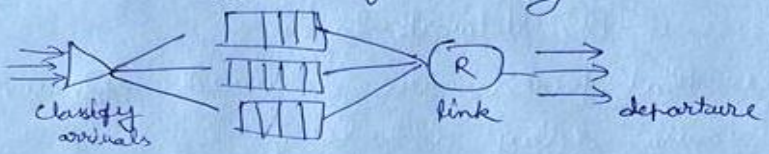
1. First Come First Serve (FCFS)/First-in-first out (FIFO): In this algorithm packets are delivered in the order of their arrival to output port. The first packet that arrives is transmitted first.



2. Priority Scheduling: In this algorithm, the arriving packets are classified by class and are queued. Packets with highest priority are transmitted before packets with lower priority.



3. Round Robin scheduling : In this algorithm arriving packets are classified and queued by class.
→ any header fields can be used for classification
→ Server, cyclically scans class queues, sending one packet from each class if available in each turn.

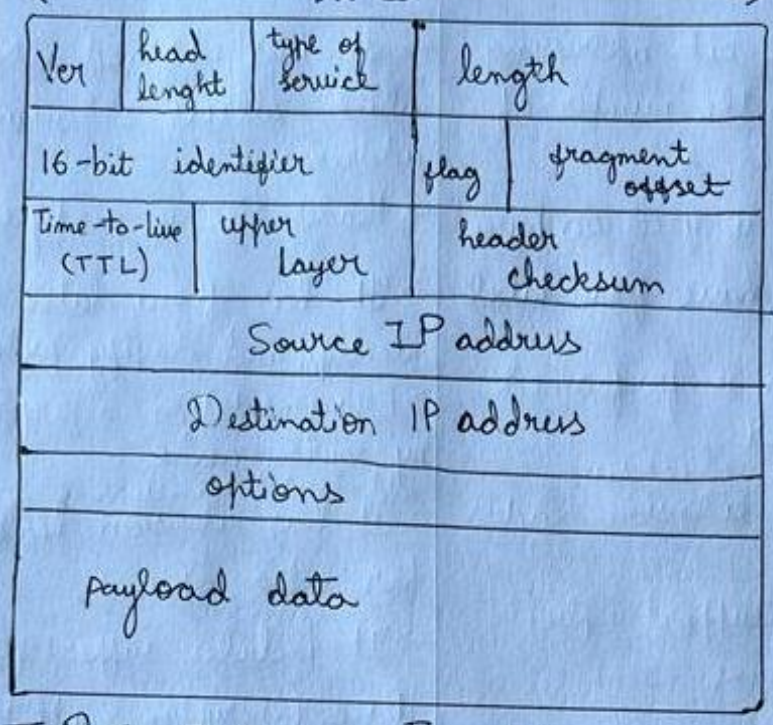


4. Weighted fair queueing (WFQ) : The packets are assigned weights based on their priority and transmitted in a weighted round robin way. Packets with higher weight are transmitted frequently than with lower weights.
→ Each class i, has weight w_i and gets weighted amount of service in each cycle $\frac{w_i}{\sum_j w_j}$
→ minimum bandwidth guarantee for each class.

Importance of Packet scheduling in Network layer :
→ It ensures that network resources are used efficiently and fairly
→ Without packet scheduling network devices can be congested and delayed.
→ Packet loss can happen without proper scheduling.
→ It improves network performance.

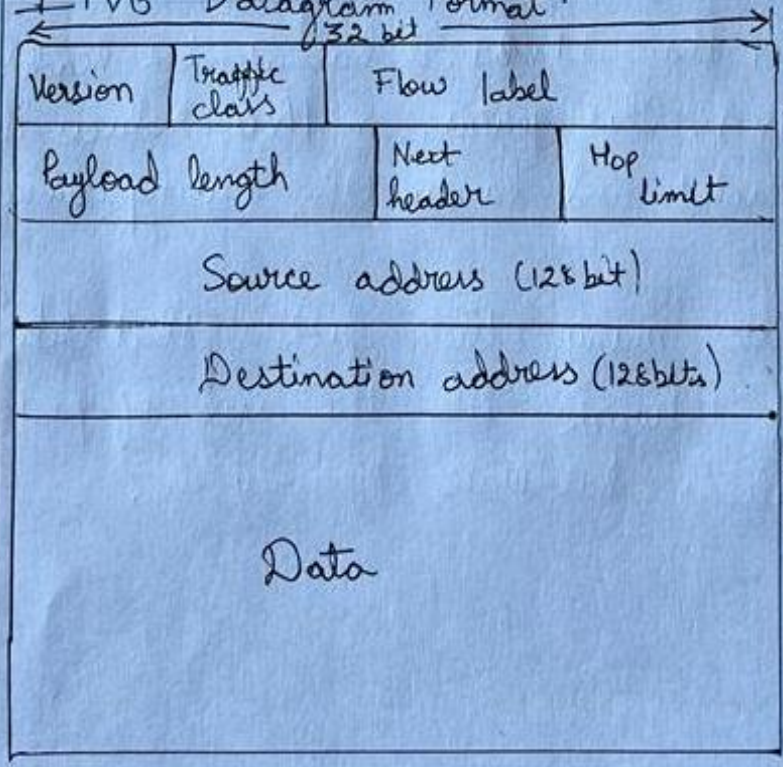
2. Give the IP datagram format for IPv4 and IPv6. What is the difference between IPv4 and IPv6 with structures? Which is faster and why? (5 points)

2. IPv4 Datagram Format:



Version : 4 bit
Header length : 4 bit
Type service : 8 bit
Total length : 16 bit
Identifier : 16 bit
Flag : 3 bit
Fragment offset : 13 bit
TTL : 8 bit
Upper layer : 8 bit
Header checksum : 16 bit

IPv6 Datagram Format:



Version: 4 bit
Traffic class : 8 bit
Flow label : 20 bit
Payload length : 16 bit
Next header : 8 bit
Hop limit : 8 bit
Source addres : 128 bit
Destination addres : 128 bit

IPv4	IPv6
→ This uses 32-bit addressing which limits the number of unique address to 2^{32}	→ It uses 128-bit addressing which allows 2^{128} unique addresses.
→ Header size is fixed 20 bytes	→ Header size is 40 bytes.
→ It does not have flow label.	→ It has flow label to identify and classify packets
→ Fragmentation is performed by the router	→ Fragmentation is performed by source node
→ There is checksum	→ It has no checksum
→ There is no extension headers	→ It has extension headers for extra options.
→ It doesnot support explicit security concerns	→ It includes support for IPsec, Encryption & authentication

→ IPv6 is faster than IPv4, it has extended headers larger than IPv4. This feature in IPv6 reduces the overhead of packet and bandwidth, making connection faster.

→ There are also many other factors that determine the speed of network like the size of packet, software abilities, hardware efficiency, etc.

3. What is NAT? Explain in detail how it works with changes in addresses in the router. (5 points)

3. Network Addressing Translation:

It is used to allow devices on a local network to access the internet using a single public IP address.

→ It changes the IP of outgoing packets, allowing multiple devices to share single public IP address.

→ all datagrams leaving local network have same source NAT IP address. but different source port numbers.

→ all devices in local network have 32-bit addresses in a private IP address space that can only be used in local network.

→ When a response is received from internet, the router uses the unique identifier added to packet to determine which device the response is intended for.

→ The router replaces destination IP with packets original IP and forwards the packet to device on local network.

NAT with change in addresses in the router:

→ A device on local network sends a packet to other device on internet.

→ The router when receives the packet, replaces source with the own public IP address.

→ It also adds unique identifier to track the device sending the packet.

→ The the router forwards this packet on the internet.

→ When the response is received, the router uses the unique identifier added to the packet to find out the device the response has to be sent to.

→ The router now replaces the destination IP with the original IP of the device.

→ It finally sends the packet to the local device.

→ If dynamic NAT is used, the router assigns a different public IP address to each device on the local network.

4. Explain the Tunneling and encapsulation mechanism in IPv6? (5 points)

4 Tunneling & Encapsulation: It is a way to use an existing IPv4 infrastructure to carry IPv6 traffic.

→ IPv6 or IPv4 hosts and routers can tunnel IPv6 datagrams over regions of IPv4 routing topology by encapsulating them within IPv4 packets.

→ IPv6 datagram is encapsulated as payload in a IPv4 datagram.

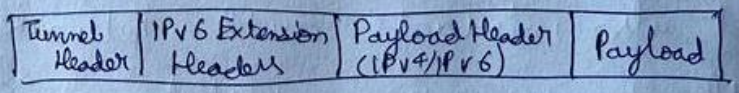
→ It allows traversing network that do not support the desired protocol.

→ Manual and automatic tunnels are the most common techniques.

→ Manual tunneling is explicitly configured at network node.

→ Automatic tunneling is created by certain OS.

IPv6 Tunnel Encapsulated Packet



5. What is software-defined networking, and What are the benefits of software-defined networking? (5 Points)

5 Software defined Network (SDN):

It is an approach that uses software-based controllers or application programming interfaces (APIs) to communicate with underlying hardware infrastructure and direct traffic.

→ It consists of 3 main parts as part of SDN architecture.

1. Dataplane switches:

→ It consists of fast, simple, commodity switches implementing generalized data plane forwarding in hardware

→ The flow tables are computed, installed under controller supervision.

→ API for switch control are used. Ex: Openflow

→ Uses these protocols to communicate with controller

2. SDN Controller

→ It is part of control plane, maintaining network state information.

→ It also interacts with network control applications using northbound API's.

→ It interacts with network switches below using southbound APIs.

3. Network Control Applications:

→ "brains of control": It implements control functions using lower-level services, API is provided by controller.

→ unbundled: can be provided by 3rd party.

Advantages:

→ SDN is critical in 5G cellular networks & cloud applications.

→ It supports moving workload around a network quickly.

→ It make the network flexible and scalable.

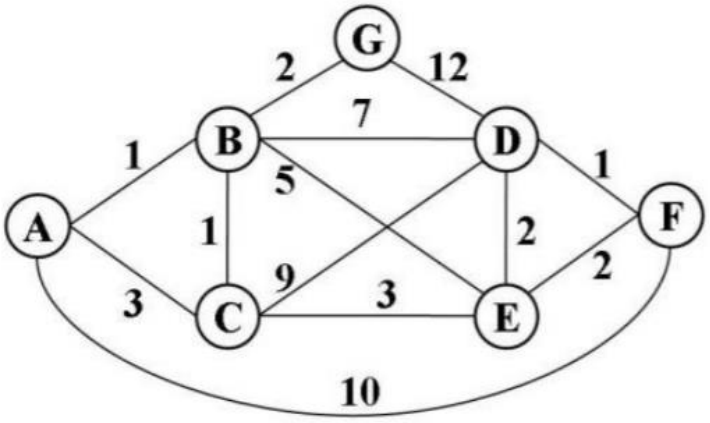
→ It has capacity to support emerging technologies like IOT & edge computing.

→ It has high performance & robustness to failure.

→ It is also secure and easy to manage.

6. Consider the following network. With the indicated link costs, use Dijkstra's shortest-path algorithm to compute the shortest path to all the nodes. Show how the algorithm works by computing a table below (Look slides for table Example). (10 Points)

FIRST NAME STARTS WITH	SHORTEST PATH FROM
A-E	A
F-J	B
K-O	C
P-T	D
U-W	E
X-Z	F

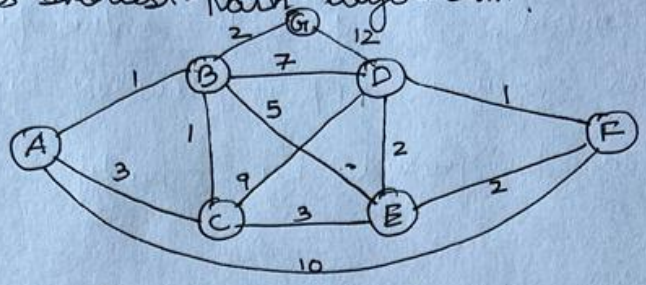


Below is just an example of first name starting with A-E

Step	N'	D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
1	A						

6. Dijkstra's shortest-path algorithm.

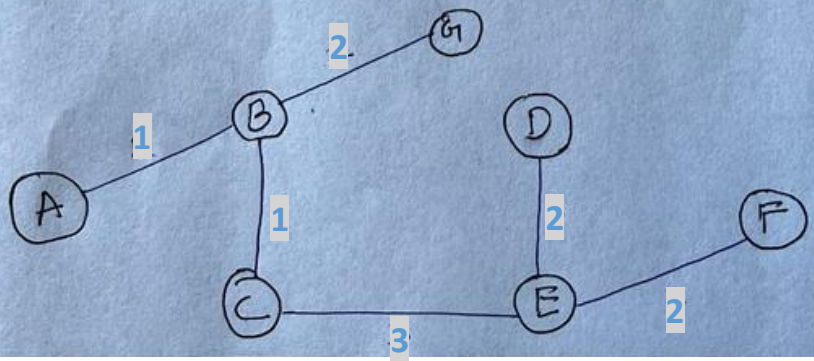
As my name starts with N, I shall start with node 'C'



$D(b) = \min\left(D(b), D(a) + C_{a,b}\right)$

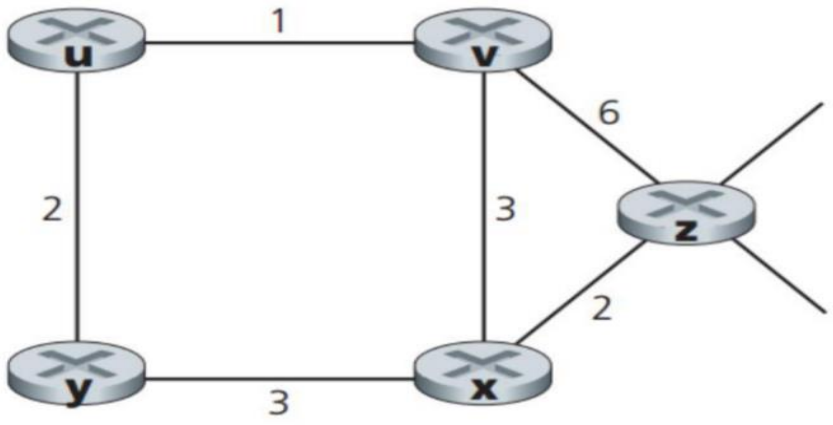
Step	N'	D(A),p(A)	D(B),p(B)	D(D),p(D)	D(E),p(E)	D(F),p(F)	D(G),p(G)
0	C	3,C	(1,C)	9	3,C	∞	∞
1	CB	(2,B)		8,B	3,C	∞	3,B
2	CBA			8,B	(3,C)	13,A	3,B
3	CBAE			5,E		5,E	(3,B)
4	CBAEG			(5,E)		5,E	
5	CBAEGD					(5,E)	
6	CBAEGDF						

Resulting least-cost path tree from 'C':



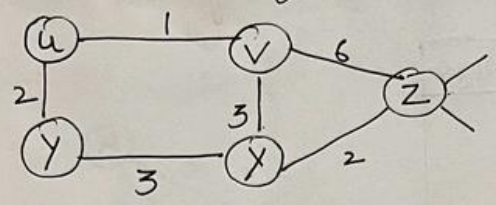
7. Consider the network shown below and assume that each node initially knows the costs to each of its neighbors. Consider the distance-vector algorithm and show the distance table entries at: (10 Points)

FIRST NAME STARTS WITH	SHORTEST PATH FROM
A-E	U
F-J	V
K-O	Y
P-T	V
U-Z	Z



Method 1:

7. Distance vector Algorithm



As my first name starts with 'N', I will start with node 'Y'.

Bellman-Ford Equation:

$D_x(Y) = \min_v \{C_{x,v} + D_v(y)\}$

Initially Distance-vector at Y

DV in y

$D_y(u) = 2$
$D_y(v) = \infty$
$D_y(x) = 3$
$D_y(y) = 0$
$D_y(z) = \infty$

Final Dv in Y	Next hop router
$D_y(u) = 2$	u
$D_y(v) = 3$	u
$D_y(x) = 3$	x
$D_y(y) = 0$	y
$D_y(z) = 5$	x

Distance vector of neighbours of Y are of 'u' & x nodes

Dv in u:

$D_u(u) = 0$
$D_u(v) = 1$
$D_u(x) = \infty$
$D_u(y) = 2$
$D_u(z) = \infty$

Dv in x:

$D_x(u) = \infty$
$D_x(v) = 3$
$D_x(x) = 0$
$D_x(y) = 3$
$D_x(z) = 2$

The neighbours of u & x are v & z

DV in v

$D_v(u) = 1$
$D_v(v) = 0$
$D_v(x) = 3$
$D_v(y) = \infty$
$D_v(z) = 6$

Dv in z

$D_z(u) = \infty$
$D_z(v) = 6$
$D_z(x) = 2$
$D_z(y) = \infty$
$D_z(z) = 0$

The final Distance vector at Y

Dv in Y

$D_y(u) = \min\{C_{y,u} + D_u(u), C_{y,x} + D_x(u)\} = \min\{2+0, 3+\infty\} = 2$ through u

$D_y(v) = \min\{C_{y,u} + D_u(v), C_{y,x} + D_x(v)\} = \min\{2+1, 3+3\} = 3$ through u

$D_y(x) = \min\{C_{y,u} + D_u(x), C_{y,x} + D_x(x)\} = \min\{2+\infty, 3+0\} = 3$ through x

$D_y(y) = 0$

$D_y(z) = \min\{C_{y,u} + D_u(z), C_{y,x} + D_x(z)\} = \min\{2+\infty, 3+2\} = 5$ through x

Method 2

My name starts with “N” hence I am going to find distance table entries at Y

The routing table at Y initially will be :

Network	**Cost**	**Next router**
u	2	u
v	∞	-
x	3	x
z	∞	-
y	0	y

Z's updated table after receiving x and u table can be calculated by first calculating u and x initial tables :

u's initial table will be:

Network	Cost	Next router
u	0	u
v	1	v
x	∞	-
z	∞	-
y	2	y

x's initial table will be :

Network	Cost	Next router
u	∞	-
v	3	v
x	0	x
z	2	z
y	3	y

To calculate Y's updated table:

Y will receive the distance - vector from u and x as :

Distance from y to u

Min { y to u, y to x -> x to v -> v to u}

Min { 2 , 3 + 3 +1} = Min { 2 , 7}

= 2

Distance from y to v

Min { y to u -> u to v, y to x -> x to v }

Min { 2 + 1 , 3 + 3 } = Min { 3 , 6}

= 3

Distance from y to x

Min { y to x, y to u -> u to v -> v to x}

Min { 3, 2 + 1 + 3 } = Min { 3 , 6}

= 3

Distance from y to z

Min { y to x -> x to z, y to u -> u to v -> v to z}

Min { 3 + 2, 2 + 1 + 6 } = Min { 5 , 9}

= 5

Distance from y to y is 0

So final distance routing table of y after receiving distances from u and x will be:

Network	Cost	Next router
u	2	u
v	3	u
x	3	x
z	5	x
y	0	y

8. Consider a datagram network using 32-bit host addresses. Suppose a router has four links, numbered 0 through 3, and packets are to be forwarded to the link interfaces as follows: (5 Points)

Destination Address Range	Range Outgoing Link Interface
11110000 00000000 00000000 00000000 through 11110000 00111111 11111111 11111111	3
11100000 01000000 00000000 00000000 through 11100000 01000001 11111111 11111111	2
11100000 01000010 00000000 00000000 through 11100001 01111111 11111111 11111111	1
Otherwise	0

a. Provide a forwarding table that has four entries, uses the longest prefix matching, and forward packets to the correct link interfaces.

Destination Address Range (longest prefix matching)	Outgoing Link Interface
11110000 00****** ******** ********	3
11100000 0100000* ******** ********	2
1110000* ******** ******** ********	1
Otherwise	0

b. Describe how your forwarding table determines the appropriate link interface for datagrams with destination addresses:

11100001 01000000 11000011 00111100

link interface: 1

11100001 11110000 00010001 01110111

link interface: 0

11110000 00010001 01010001 01010101

link interface: 3

11100000 01000000 00010000 00100100

link interface: 2

00000000 00000000 00000000 00000000

link interface: 0

c. Rewrite this forwarding table using the a.b.c.d/x notation instead of the binary string notation.

Destination Address Range (longest prefix matching)	Outgoing Link Interface
240.0.0.0/10	3
224.64.0.0/16	2
224.0.0.0/8	1
225.128.0.0/9	0
Otherwise	0

The above-highlighted row can be excluded.