CMR
INSTITUTE OF
TECHNOLOGY

USN

Internal Assessment Test 2 – November 2016

Sub:	Computer Networks							Code:	13MCA 31
Date:	*2/11/2016*	*Duration:*	*90 mins*	*Max Marks:*	*50*	*Sem:*	*3*	*Branch:*	*MCA*

Answer any five of the following. 5 x 10=50M

1. Explain link state routing algorithm in detail. (10 Marks)

- It is adynamic routing algorithm
- The idea behind link state routing can be stated as five parts. Each router must do the following:

1.Discover its neighbors and learn their network addresses.

2.Measure the delay or cost to each of its neighbors.

3.Construct a packet telling all it has just learned.

4.Send this packet to all other routers.

5.Compute the shortest path to every other router.

The complete topology and all delays are experimentally measured and distributed to every router. Then Dijkstra's algorithm can be run to find the shortest path to every other router

1. Learning about the Neighbors

- It accomplishes this goal by sending a special HELLO packet on each point-to-point line. The router on the other end is expected to send back a reply telling who it is. These names must be globally unique because when a distant router later hears that three routers are all connected to F, it is essential that it can determine whether all three mean the same F.

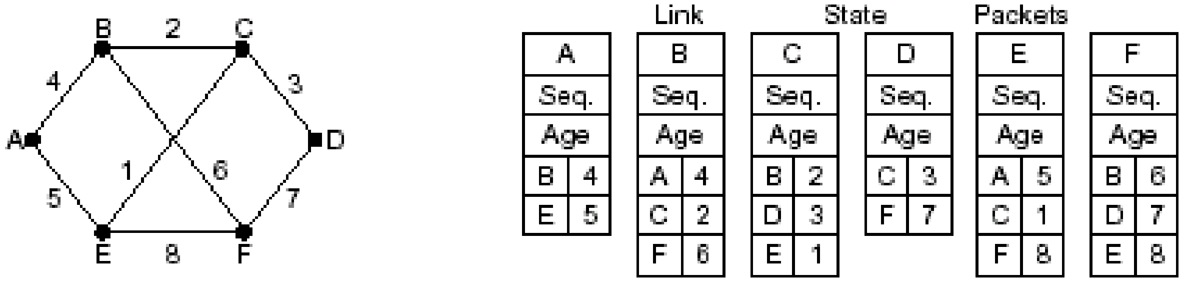
2. Measuring Line Cost

- The most direct way to determine this delay is to send over the line a special ECHO packet that the other side is required to send back immediately. By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the delay. For even better results, the test can be conducted several times, and the average used. Of course, this method implicitly assumes the delays are symmetric, which may not always be the case.

3. Building Link State Packets

- The packet starts with the identity of the sender, followed by a sequence number and age (to be described later), and a list of neighbors. For each neighbor, the delay to that neighbor is given.

(a) A subnet. (b) The link state packets for this subnet



Link State Packets

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(a) (b)

- Building the link state packets is easy. The hard part is determining when to build them. One possibility is to build them periodically, that is, at regular intervals. Another possibility is to build them when some significant event occurs, such as a line or neighbor going down or coming back up again or changing its properties appreciably.

4. Distributing the Link State Packets

- The basic distribution algorithm:The fundamental idea is to use flooding to distribute the link state packets. To keep the flood in check, each packet contains a sequence number that is incremented for each new packet sent. Routers keep track of all the (source router, sequence) pairs they see. When a new link state packet comes in, it is checked against the list of packets already seen. If it is new, it is forwarded on all lines except the one it arrived on. If it is a duplicate, it is discarded. If a packet with a sequence number lower than the highest one seen so far ever arrives, it is rejected as being obsolete since the router has more recent data.

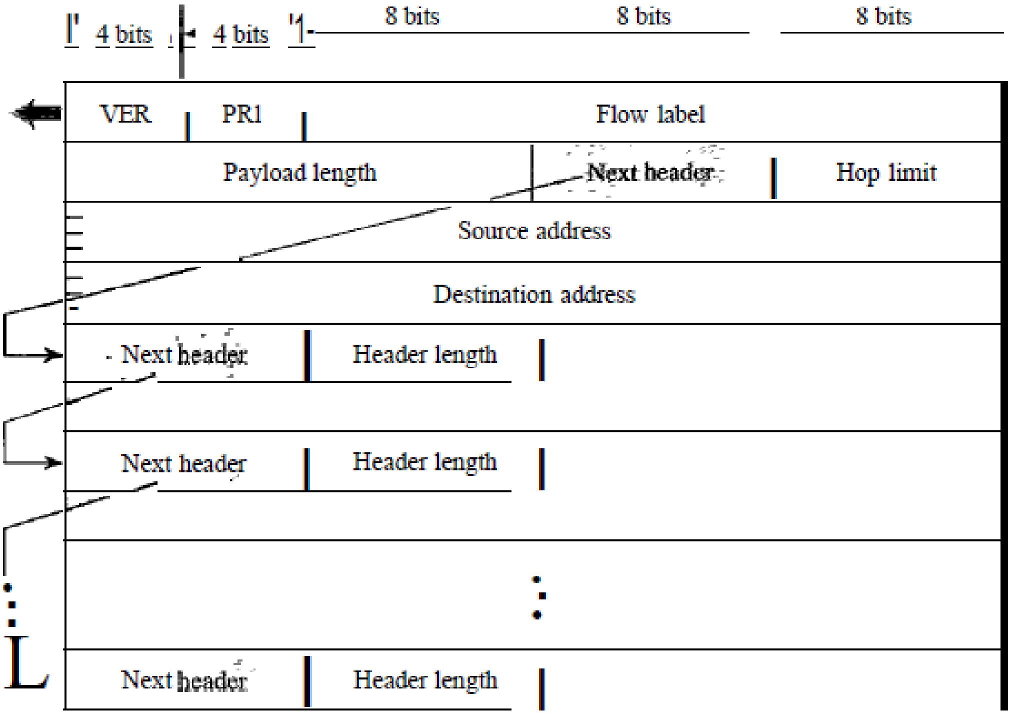
5. Computing the New Routes

- Once a router has accumulated a full set of link state packets, it can construct the entire subnet graph because every link is represented.

Now Dijkstra's algorithm can be run locally to construct the shortest path to all possible destinations

2. Explain IPV6 header format with a neat diagram. (10 Marks)

Each packet is composed of a mandatory base header followed by the payload. The payload consists of two parts: optional extension headers and data from an upper layer. The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65,535 bytes of information.
Base Header
Header shows the base header with its eight fields.



These fields are as follows:

Version: This 4-bit field defines the version number of the IP. For IPv6, the value is 6.
Priority: The 4-bit priority field defines the priority of the packet with respect to traffic congestion.

Flow label: The flow label is a 3-byte (24-bit) field that is designed to provide special handling for a particular flow of data. We will discuss this field later.
Payload length: The 2-byte payload length field defines the length of the IP datagram excluding the base header.
Next header: The next header is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP. Each extension header also contains this field.
Hop limit: This 8-bit hop limit field serves the same purpose as the TIL field in IPv4.
Source address: The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.
Destination address: The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram. If source routing is used, this field contains the address of the next router.

Extension Headers:

The length of the base header is fixed at 40 bytes. However, to give greater functionality to the IP datagram, the base header can be followed by up to six extension headers

3. What is traffic shaping? Explain Leaky bucket algorithm. (10 Marks)

Traffic shaping is a technique for regulating the average rate and burstiness of a flow of data that enters the network. The goal is to allow applications to transmit a wide variety of traffic that suits their needs, including some bursts, yet have a simple and useful way to describe the possible traffic patterns to the network. When a flow is set up, the user and the network (i.e., the customer and the provider) agree on a certain traffic pattern (i.e., shape) for that flow. In effect, the customer says to the provider “My transmission pattern will look like this; can you handle it?’

Traffic shaping reduces congestion and thus helps the network live up to its promise. However, to make it work, there is also the issue of how the provider can tell if the customer is following the agreement and what to do if the customer is not. Packets in excess of the agreed pattern might be dropped by the network, or they might be marked as having lower priority. Monitoring a traffic flow is called

traffic policing.

The rate at which water enters the bucket, the outflow is at a constant rate, *R*, when there is any water in the bucket and zero when the bucket is empty. Also, once the bucket is full to capacity *B*, any additional water entering it spills over the sides and is lost.

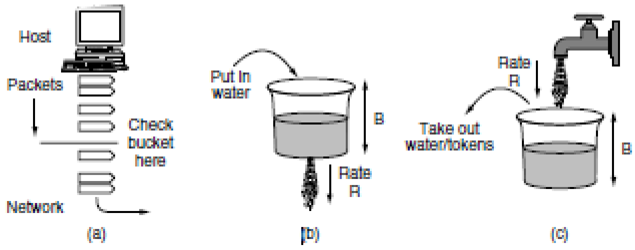


Figure 5-28. (a) Shaping packets. (b) A leaky bucket. (c) A token bucket.

This bucket can be used to shape or police packets entering the network, as shown in Fig. 5-28(a). Conceptually, each host is connected to the network by an interface containing a leaky bucket. To send a packet into the network, it must be possible to put more water into the bucket. If a packet arrives when the bucket is full, the packet must either be queued until enough water leaks out to hold it or be discarded. The former might happen at a host shaping its traffic for the network as part of the operating system. The latter might happen in hardware at a provider network interface that is policing traffic entering the network. This technique was proposed by Turner (1986) and is called the **leaky bucket algorithm**.

Leaky Bucket Algorithm is the most commonly used policing mechanism o Bucket has specified leak rate for average contracted rate o Bucket has specified depth to accommodate variations in arrival rate o Arriving packet is conforming if it does not result in overflow Leaky Bucket algorithm can be used to police arrival rate of a packet stream

The above figure shows the leaky bucket algorithm that can be used to police the traffic flow

At the arrival of the first packet, the content of the bucket is set to zero and the last conforming time (LCT) is set to the arrival time of the first packet.

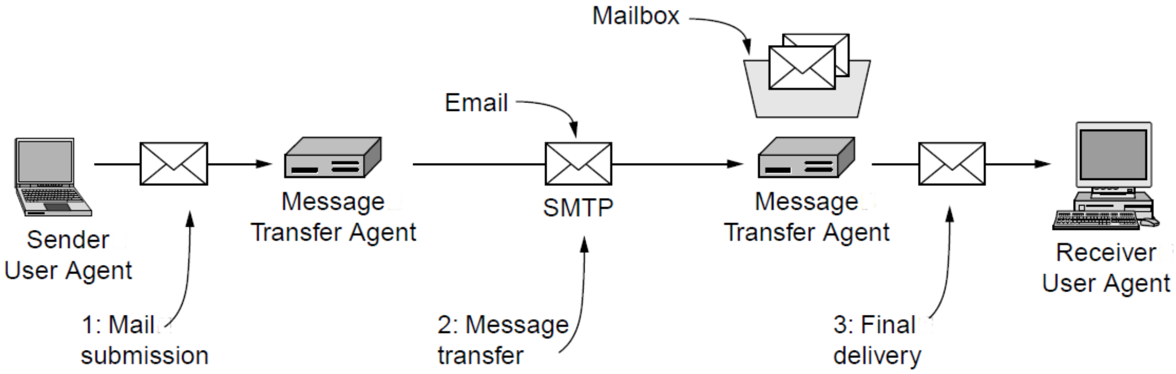
The depth of the bucket is L+I, where l depends on the traffic burstiness.

At the arrival of the kth packet, the auxiliary variable X’ records the difference between the bucket content at the arrival of the last conforming packet and the interarrival time between the last conforming packet and the kth packet.

If the auxiliary variable is greater than L, the packet is considered as nonconforming, otherwise the packet is conforming. The bucket content and the arrival time of the packet are then updated.

4. Explain the architecture of email system. (10 Marks)

The architecture of the email system is shown inFig. It consists of two kinds of subsystems: the **user agents**, which allowpeople to read and send email, and the **message transfer agents**, which move the messages from the source to the destination. We will also refer to message transfer agents informally as **mail servers**.



The user agent is a program that provides a graphical interface, or sometimes a text- and command-based interface that lets users interact with the email system. It includes a means to compose messages and replies to messages, display incoming messages, and organize messages by filing, searching, and discarding them. The act of sending new messages into the mail system for delivery is called **mail submission**.

Some of the user agent processing may be done automatically, anticipating what the user wants. For example, incoming mail may be filtered to extract or deprioritize messages that are likely spam. Some user agents include advanced features, such as arranging for automatic email .A user agent runs on the same computer on which a user reads her mail. It is just another program and may be run only some of the time.

The message transfer agents are typically system processes. They run in the background on mail server machines and are intended to be always available.Their job is to automatically move email through the system from the originator to the recipient with **SMTP** (**Simple Mail Transfer Protocol**). This is the message transfer step.

Message transfer agents also implement **mailing lists**, in which an identical copy of a message is delivered to everyone on a list of email addresses. Other advanced features are carbon copies, blind carbon copies, high-priority email, secret (i.e., encrypted) email, alternative recipients if the primary one is not currently available, and the ability for assistants to read and answer their bosses' email. Linking user agents and message transfer agents are the concepts of mailboxes and a standard format for email messages. **Mailboxes** store the email that is received for a user. They are maintained by mail servers. User agents simply present users with a view of the contents of their mailboxes. To do this, the user agents send the mail servers commands to manipulate the mailboxes, inspecting their contents, deleting messages, and so on. The retrieval of mail is the final delivery (step 3) in Fig. With this architecture, one user may use different user agents on multiple computers to access one mailbox.Mail is sent between message transfer agents in a standard format.

5. Explain the following.

a) DNS (5 Marks)

Uses transport services to build distributed applications. For the Internet, the top of the naming hierarchy is managed by an organization called **ICANN** (**Internet Corporation for Assigned Names and Numbers**).

The Internet is divided into over 250 **top-level domains**, where each domain covers many hosts. Each domain is partitioned into sub domains, and these are further partitioned, and so on. The leaves of the tree represent domains that have no subdomains (but do contain machines, of course). A leaf domain may contain a single host, or it may represent a company and contain thousands of hosts.

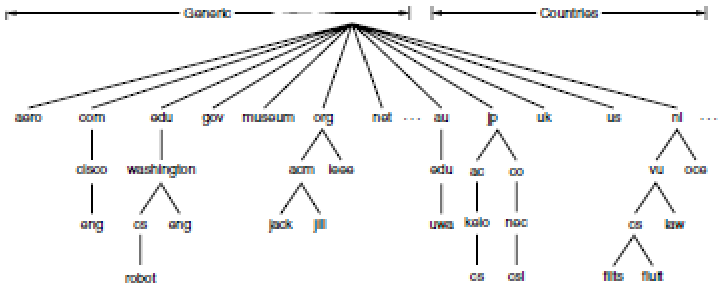


Figure 7-1. A portion of the Internet domain name space.

The top-level domains are run by **registrars** appointed by ICANN. Getting a name merely requires going to a corresponding registrar (for *com* in this case) to check if the desired name is available and not somebody else's trademark. If there are no problems, the requester pays the registrar a small annual fee and gets the name.

Every domain, whether it is a single host or a top-level domain, can have a set of **resource records** associated with it. These records are the DNS database. For a single host, the most common resource record is just its IP address, but many other kinds of resource records also exist. When a resolver gives a domain name to DNS, what it gets back are the resource records associated with that name. Thus, the primary function of DNS is to map domain names onto resource records.

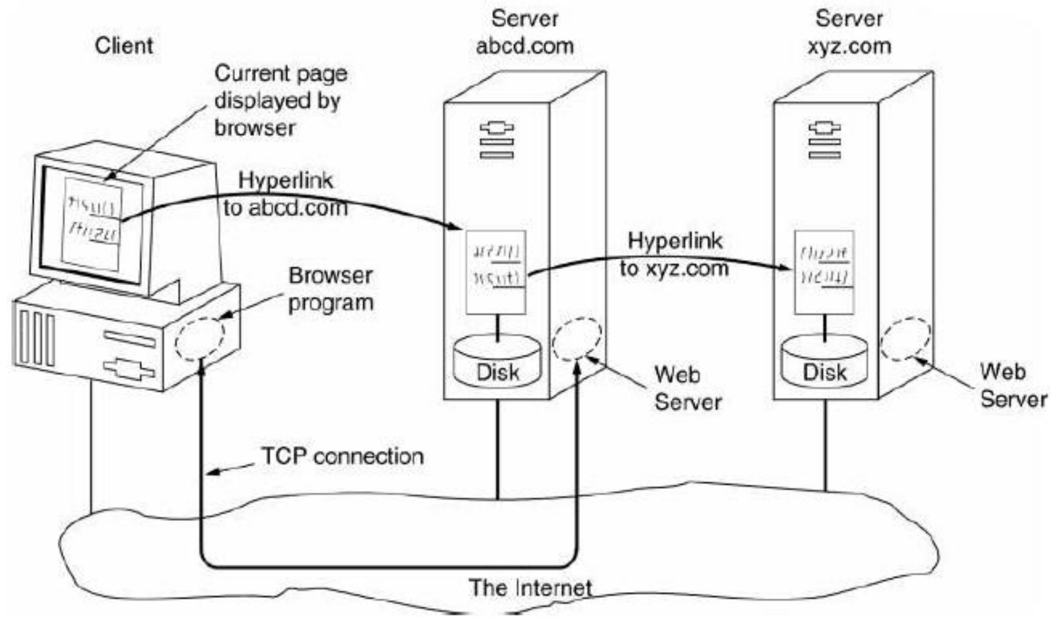
A resource record is a five-tuple. Although they are encoded in binary for efficiency, in most expositions resource records are presented as ASCII text, one line per resource record. The format we will use is as follows:

Domain name Time to live Class Type Value

b) WWW (5 Marks)

The Web, as the World Wide Web is popularly known, is an architectural framework for accessing linked content spread out over millions of machines all over the Internet.

The Web consists of a vast, worldwide collection of content in the form of **Web pages**, often just called **pages** for short. Each page may contain links to other pages anywhere in the world. Users can follow a link by clicking on it, which then takes them to the page pointed to. This process can be repeated indefinitely. The idea of having one page point to another, now called **hypertext**.



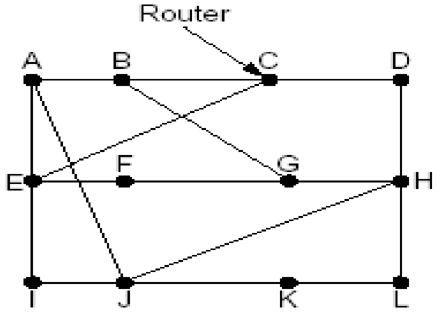
The parts of the Web model.

Steps a client (browser) takes to follow a hyperlink:
− Determine the protocol (HTTP)
− Ask DNS for the IP address of server
− Make a TCP connection to server
− Send request for the page; server sends it back
− Fetch other URLs as needed to display the page
− Close idle TCP connections
Steps a server takes to serve pages:
− Accept a TCP connection from client
− Get page request and map it to a resource (e.g., file name)
− Get the resource (e.g., file from disk)
− Send contents of the resource to the client.
− Release idle TCP connections

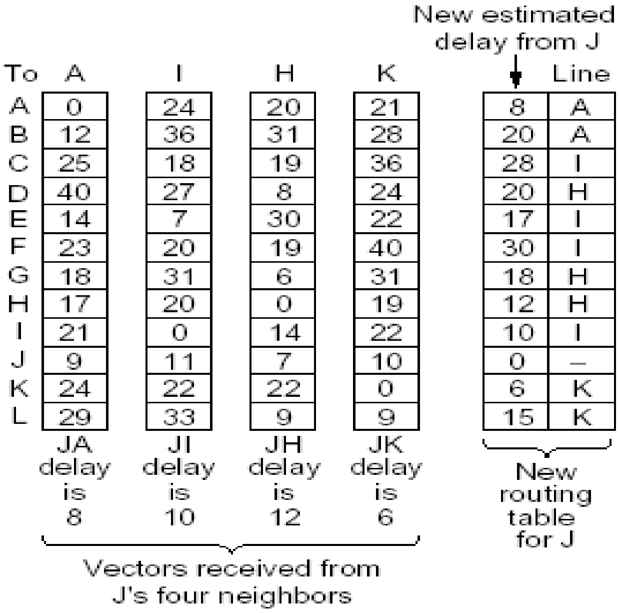
6. a) Explain Distance vector routing algorithm. (5 Marks)

- A dynamic routing algorithm
- **Distance vector routing algorithms** operate by having each router maintain a table (i.e, a vector) giving the best known distance to each destination and which line to use to get there. These tables are updated by exchanging information with the neighbors. (also named the distributed Bellman-Ford routing algorithm and the Ford-Fulkerson algorithm)
- **Table content:** In distance vector routing, each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. This entry contains two parts: the preferred outgoing line to use for that destination and an estimate of the time or distance to that destination.
- **Table updating method**:Assume that the router knows the delay to each of its neighbors. Once every T msec each router sends to each neighbor a list of its estimated delays to each destination. It also receives a similar list from each neighbor. Imagine that one of these tables has just come in from neighbor X, with X_i being X's estimate of how long it takes to get to router i. If the router knows that the delay to X is m msec, it also knows that it can reach router i via X in X_i + m msec. By

performing this calculation for each neighbor, a router can find out which estimate seems the best and use that estimate and the corresponding line in its new routing table. Note that the old routing table is not used in the calculation.



(a)


New estimated delay from J
Vectors received from J's four neighbors
New routing table for J

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6		

(b)

Part (a) shows a subnet. The first four columns of part (b) show the delay vectors received from the neighbors of router J. Suppose that J has measured or estimated its delay to its neighbors, A, I, H, and K as 8, 10, 12, and 6 msec, respectively

b) Explain UDP header format. (5 Marks)

The Internet protocol suite supports a connectionless transport protocol called **UDP** (**User Datagram Protocol**). UDP provides a way for applications to send encapsulated IP datagrams without having to establish a connection.UDP transmits **segments** consisting of an 8-byte header followed by the payload.

The header is shown in Fig. The two **ports** serve to identify the endpoints within the source and destination machines. When a UDP packet arrives,its payload is handed to the process attached to the destination port. This attachment occurs when the BIND primitive or something similar is used.Without the port fields, the transport layer would not know what to do with each incoming packet. With them, it delivers the embedded segment to the correct application.

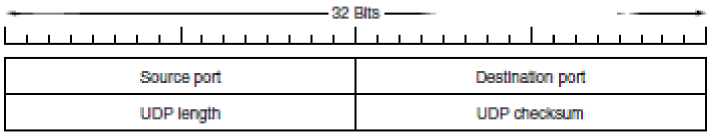


Figure 6-27. The UDP header.

The source port is primarily needed when a reply must be sent back to the source. By copying the *Source port* field from the incoming segment into the *Destination port* field of the outgoing segment, the process sending the reply can specify which process on the sending machine is to get it.

The *UDP length* field includes the 8-byte header and the data. The minimum length is 8 bytes, to cover the header. The maximum length is 65,515 bytes, which is lower than the largest number that will fit in 16 bits because of the size limit on IP packets.

An optional *Checksum* is also provided for extra reliability. It checksums the header, the data, and a conceptual IP pseudoheader. When performing this computation, the *Checksum* field is set to zero and the data field is padded out with an additional zero byte if its length is an odd number. The checksum algorithm is simply to add up all the 16-bit words in one's complement and to take the one's complement of the sum. As a consequence, when the receiver performs the calculation on the entire segment, including the *Checksum* field, the result should be 0.If the checksum is not computed, it is stored as a 0, one’s complement arithmetic a true computed 0 is stored as all 1s).

7. a) What do you mean by subnetting? (5 Marks)

We can allow the block of addresses to be split into several parts for internal use as multiple networks, while still acting like a single network to the outside world. This is called **subnetting** and the networks (such as Ethernet LANs) that result from dividing up a larger network are called **subnets.** We should be aware that this new usage of the term conflicts with older usage of ‘‘subnet’’ to mean the set of all routers and communication lines in a network.
The single /16 has been split into pieces. This split does not need to be even, but each piece must bealigned so that any bits can be used in the lower host portion. In this case, half of the block (a /17) is allocated to the Computer Science Dept, a quarter is allocated to the Electrical Engineering Dept. (a /18), and one eighth (a /19) to the Art Dept. The remaining eighth is unallocated. A different way to see how the block was divided is to look at the resulting prefixes when written in binary notation:
Computer Science: 10000000 11010000 1|xxxxxxx xxxxxxxx
Electrical Eng.: 10000000 11010000 00|xxxxxx xxxxxxxx
Art: 10000000 11010000 011|xxxxx xxxxxxxx
Here, the vertical bar (|) shows the boundary between the subnet number and the host portion.

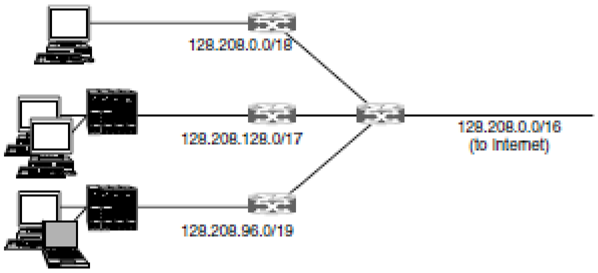


Figure 5-49. Splitting an IP prefix into separate networks with subnetting.

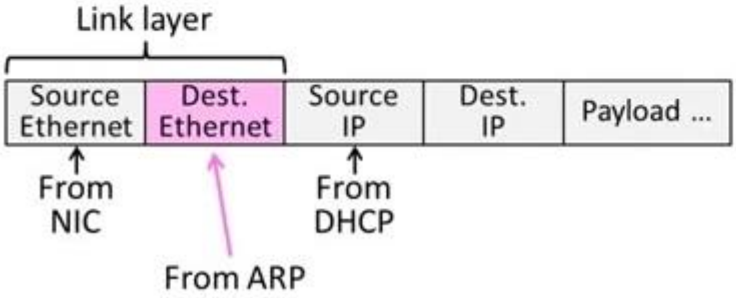
When a packet arrives, the router looks at the destination address of the packet and checks which subnet it belongs to. The router can do this by ANDing the destination address with the mask for each subnet and checking to see if the result is the corresponding prefix.

b) Explain ARP.   (5 Marks)

Even though every machine on the Internet has one or more IP addresses, these addresses are not sufficient for sending packets. Data link layer NICs such as Ethernet cards do not understand Internet addresses. They send and receive frames based on 48-bit Ethernet addresses. To facilitate this process of mapping of addresses, a protocol named ARP, or Address Resolution Protocol, is used.

To keep the cached information current and to optimize performance is to have every machine broadcast its mapping when it is configured. This broadcast is generally done in the form of an ARP looking for its own IP address. We have to make or update an entry in everyone’s ARP cache.This is known as a **gratuitous ARP**

- Node uses to map a local IP address to its Link layer addresses


Link layer
Source Ethernet
Dest. Ethernet
Source IP
Dest. IP
Payload …
From NIC
From DHCP
From ARP