TRANSPORT LAYER

Multiple Choice Type Questions

1. For stop-and-wait flow control, for n data packet sent, ... acknowledgements are needed [WBUT 2017]
   a) n b) $2n$ c) $n-1$ d) $n+1$
Answer: (a)

2. Service point addressing is done in [WBUT 2017]
   a) Physical layer b) Network layer
   c) Transport layer d) Application layer
Answer: (c)

3. Which of the following is not a WAN technology? [WBUT 2018]
   a) X.25 b) Frame Relay c) TCP/IP d) None of these
Answer: (a)

4. The process-to-process delivery of the entire message is the responsibility of the ..... layer. [MODEL QUESTION]
   a) Network b) Transport c) Application d) Physical
Answer: (c)

5. The maximum size of TCP header is [MODEL QUESTION]
   a) 64 Byte b) 2^{16} Byte c) 60 Byte d) 16 Byte
Answer: (d)

6. Jitter is due to [MODEL QUESTION]
   a) large number of packets in the net
   b) long packet size
   c) variation in the delay encountered by the packet
   d) long delay encountered by the packet
Answer: (c)

7. Flow control is the responsibilities of the [MODEL QUESTION]
   a) Data link layer b) Transport layer c) Both of these d) none of these
Answer: (c)

8. 4-way handshaking of connection establishment is associated with
   a) HTTP protocol b) UDP protocol [MODEL QUESTION]
   c) TCP protocol d) FTP protocol
Answer: (c)

9. Connection establishment involves a ..... handshake. [MODEL QUESTION]
   a) one-way b) two-way c) three-way d) none of these
Answer: (c)

Short Answer Type Questions

1. Explain Leaky bucket algorithm for congestion control. [WBUT 2012]

Answer:

The leaky bucket is an algorithm used in packet switched computer networks and telecommunications networks to check that data transmissions conform to defined limits on bandwidth and burstiness (a measure of the unevenness or variations in the traffic flow). The leaky bucket algorithm is also used in leaky bucket counters, e.g. to detect when the average or peak rate of random or stochastic events or stochastic processes exceed defined limits.

The Leaky Bucket Algorithm is based on an analogy of a bucket .that has a hole in the bottom through which any water it contains will leak away at a constant rate, until or unless it is empty. Water can be added intermittently, i.e. in bursts, but if too much is added at once, or it is added at too high an average rate, the water will exceed the capacity of the bucket, which will overflow.

There are actually two different methods of applying this analogy described in the literature .These give what appear to be two different algorithms, both of which are referred to as the leaky bucket algorithm. This has resulted in confusion about what the leaky bucket algorithm is and what its properties are.

In one version, the analogue of the bucket is a counter or variable, separate from the flow of traffic, and is used only to check that traffic conforms to the limits, i.e. the analogue of the water is brought to the bucket by the traffic and added to it so that the level of water in the bucket indicates conformance to the rate and burstiness limits. This version is referred to here as the leaky bucket as a meter. In the second version, the traffic passes through a queue that is the analogue of the bucket: i.e. the traffic is the analogue of the water passing through the bucket. This version is referred to here as the leaky bucket as a queue. The leaky bucket as a meter is equivalent to (a mirror image of) the token bucket algorithm, and given the same parameters will see the same traffic as conforming or nonconforming. The leaky bucket as a queue can be seen as a special case of the leaky bucket as a meter.

2. Why transport layer is called process to process communication layer? [WBUT 2014, 2015]

Answer:

Real communication takes place between two processes (application programs) in a network. This is called process to process delivery. The transport layer is responsible for process -to -process delivery-the delivery of a packet, part of a message, from one process to another. At the transport layer we need a transport layer address, called a port number, to choose among multiple processes running on the destination host.

3. What is the maximum and minimum size of TCP header and why? [WBUT 2014, 2015]

Answer:
The minimum size of TCP header is 5 words and the maximum is 15 words thus giving the minimum size of 20 bytes and maximum of 60 bytes, allowing for up to 40 bytes of options in the header. This field gets its name from the fact that it is also the offset from the start of the TCP segment to the actual data.

4. Explain the methods of token bucket algorithm? **[WBUT 2014]**

Answer:
The token bucket is a control mechanism that dictates when traffic can be transmitted, based on the presence of tokens in the bucket. The token bucket contains tokens, each of which can represent a unit of bytes. The network administrator specifies how many tokens are needed to transmit however many number of bytes. When tokens are present, a flow is allowed to transmit traffic. If there are no tokens in the bucket, a flow cannot transmit its packets. Therefore, a flow can transmit traffic up to its peak burst rate if there are adequate tokens in the bucket and if the burst threshold is configured appropriately.
The algorithm can be conceptually understood as follows:

- A token is added to the bucket every 1 / r seconds.
- The bucket can hold at the most b tokens. If a token arrives when the bucket is full, it is discarded.
- When a packet (network layer PDU) of n bytes arrives, n tokens are removed from the bucket, and the packet is sent to the network.
- If fewer than n tokens are available, no tokens are removed from the bucket, and the packet is considered to be non-conformant.

The algorithm allows bursts of up to b bytes, but over the long run the output of conformant packets is limited to the constant rate, r. Non-conformant packets can be treated in various ways:

- They may be dropped.
- They may be enqueued for subsequent transmission when sufficient tokens have accumulated in the bucket.

They may be transmitted, but marked as being non-conformant, possibly to be dropped subsequently if the network is overloaded.

5. What are the difference between TCP and UDP? **[WBUT 2014, 2016]**

Answer:

TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
TCP is a connection-oriented protocol, a connection can be made from client to server, and from then on any data can be sent along that connection.	A simpler message-based connectionless protocol. With UDP you send messages (packets) across the network in chunks.
Reliable - when you send a message along a TCP socket, you know it will get there unless the connection fails completely. If it gets lost along the way, the server will re-request the lost part. This means complete integrity, things	**Unreliable** - When you send a message, you don't know if it'll get there, it could get lost on the way.

TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
don't get corrupted.	
Ordered - if you send two messages along a connection, one after the other, you know the first message will get there first. You don't have to worry about data arriving in the wrong order.	**Not ordered** - If you send two messages out, you don't know what order they'll arrive in.
Heavyweight - when the low level parts of the TCP "stream" arrive in the wrong order, resend requests have to be sent, and all the out of sequence parts have to be put back together, so requires a bit of work to piece together.	**Lightweight** - No ordering of messages, no tracking connections, etc. It's just fire and forget! This means it's a lot quicker, and the network card / OS have to do very little work to translate the data back from the packets.

6. What is the difference between leaky bucket and token bucket technique of QOS? **[WBUT 2016]**

Answer:

Difference between leaky bucket and token bucket algorithm:

Token Bucket	Leaky Bucket
Token dependent.	Token independent.
If bucket is full token are discarded, but not the packet.	If bucket is full packet or data is discarded.
Packets can only transmitted when there are enough token.	Packets are transmitted continuously.
It allows large burst to be sent faster rate after that constant rate.	It sends the packet at constant rate.
It saves to send large bursts.	It does not save token.

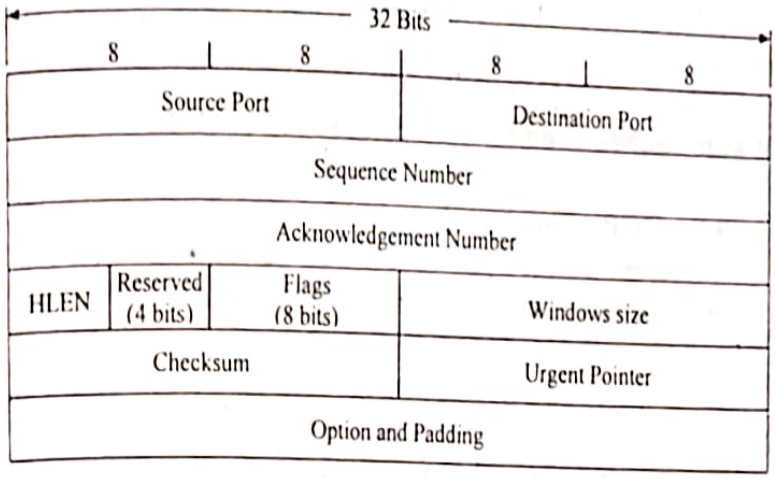
7. Describe TCP header format. **[WBUT 2016]**

OR,

Draw and explain TCP state transition diagram for connection management. **[WBUT 2018]**

Answer:

TCP attaches a header to the application layer data; the header contains fields for the sequence numbers and other information necessary for these mechanisms and fields for addresses called port numbers, which identify the source and destination application of the data. The application data with its attached TCP header is then encapsulated within an IP packet for delivery. Figure below shows the fields of the TCP header. Source and Destination Port are 16-bit fields that specify the source and destination applications for the encapsulated data.



TCP header format

Like other numbers used by TCP/IP, RFC 1700 describes all port numbers in common and not-so-common use. A port number for an application, when coupled with the IP address of the host the application resides on, is called a socket. A socket uniquely identifies every application in a network.

Sequence Number is a 32-bit number that identifies where the encapsulated data fits within a data stream from the sender. For example, if the sequence number of a segment is 1343 and the segment contains 512 octets of data, the next segment should have a sequence number of 1343 + 512 +1 = 1856.

Acknowledgement Number is a 32 bit field indentifies the sequence number the source next expects to receive from the destination. If a host receives an acknowledgement number that does not match the next sequence number it intends to send (or has sent), it knows that packets have been lost.

Header Length, sometimes called Data Offset, is a four-bit field indicating the length of the header in 32-bit words. This field is necessary to identify the beginning of the data because the length of the Options field is variable.

The Reserves field is four bits, which are always set to zero.

Flags are eight 1-bit flags that used for data flow and connection control. The flags, form left to right, are Congestion Window reduced (CWR), ECN-Echo (ECE), Urgent (URG), Acknowledgement (ACK), Push (PSH), Reset (RST), Synchronize (SYN), and Final (FIN).

Windows size is a 16-bit field used for flow control. It specifies the number of octets, starting with the octet indicated by the Acknowledgement Number, that the sender of the segment will accept from its peer at the other end of the connection before the peer must stop transmitting and wait for an acknowledgment.

Checksum is 16 bits, covering both the header and the encapsulated data, allowing error detection.

Urgent Pointer is used only when the URG flag is set. The 16-bit number is added to the Sequence Number to indicate the end of the urgent data.

Options, as the name implies, specifies options required by the sender's TCP process. The most commonly used option is Maximum Segment Size, which informs the receiver

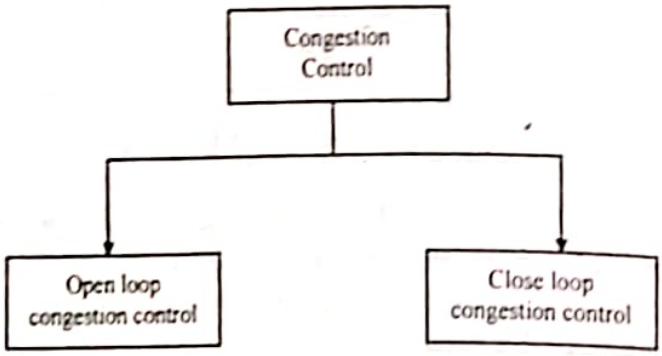
of the largest segment the sender is willing to accept. The remainder of the field is padded with zeros to ensure that the header length is a multiple of 32 octets.

8. Brief about approaches used for TCP congestion control. [WBUT 2018]

Answer:

Approaches used for TCP congestion control:

Congestion control refers to the techniques used to control or prevent congestion. Congestion control technique can be broadly classified into two categories: They are



Open Loop Congestion Control:

Open loop congestion control policies are applied to prevent congestion before it happens. The congestion control is handled either by the source or the destination. Policies adapted by open loop congestion control–

i) Retransmission Policy:

It is the policy in which retransmission of the packets are taken care.

ii) Window Policy:

The type of window at the sender side may also effect the congestion. Several packets in the Go-back-n window are resent, although some packets may be received with success at the receiver side.

iii) Discarding Policy:

A good discarding policy adopted by the routers is that the routers may prevent congestion and it also able to maintain the quality of massage. And others are–

iv) Acknowledgement Policy

v) Admission Policy

Closed Loop Congestion Control:

Closed loop congestion control technique is used to treat congestion after it happens. Several techniques are used by different protocols, some of them are Backpressure, choke packet technique, Implicit Signaling, Explicit Signaling, etc.

Long Answer Type Questions

1. Write short note QoS in Transport Layer. [WBUT 2016]

Answer:

A transport layer that provides *Quality of Service* (QoS) allows a user sender to specify the quality of transmission service desired. The protocol then presumably optimizes

network places limits on the service that can be provided by the transport protocol, a user sender should recognize two facts: (1) depending on the underlying network's capabilities, a transport protocol will have varying degrees of success in providing the requested QoS, and (2) there is a trade-off among QoS parameters such as reliability, delay, throughput, and cost of service.

Transport QoS is a broad and complicated issues. No universally accepted list of parameters exists and how the transport layer should behave for a desired QoS under different circumstances is unclear. The ISO Transport Service defines a number of possible performance QoS parameters that are negotiated during connection establishment. User senders can specify (sustained) target, acceptable and minimum values for various service parameters. The transport protocol examines these parameters and determines whether it can provide the required service; this depends in part on the available network service. ISO specifies eleven QoS parameters:

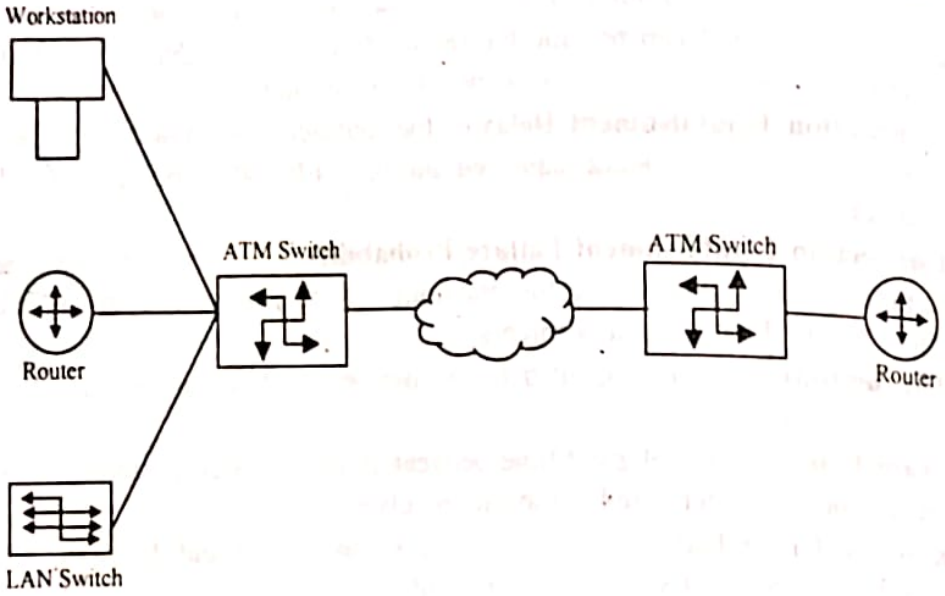
- **Connection Establishment Delay** is the maximum acceptable time between a transport connection being requested and its confirmation being received by the user sender.
- **Connection Establishment Failure Probability** is the probability a connection cannot be established within the maximum connection establishment delay time due to network or internal problems.
- **Throughput** is the number of bytes of user sender data transferred per unit time over some time interval.
- **Transit Delay** is the elapsed time between a message being submitted by a user sender and being delivered to the user receiver.
- **Residual Error Rate** is the ratio of interconnect, lost and duplicate TSDUs to the total number of TSDUs that were sent.
- **Transfer Failure Probability** is the ratio of total transfer failures to total transfer samples observed during a performance measurement.
- **Connection Release Delay** is the maximum acceptable time between a transport user initiating release of a connection and the actual release at the peer transport service user.
- **Connection Release Failure Probability** is the fraction of connection release attempts that did not complete within the agreed upon connection release delay interval.
- **Protection** is used by the user sender to specify interest in having the transport protocol provide protection against unauthorized third parties reading or modifying the transmitted data.
- **Priority** allows a user sender to specify the relative importance of transport connections. In case of congestion or the need to recover resources, lower-priority connections are degraded or terminated before the higher-priority ones.
- **Resilience** is the probability that the transport protocol itself will spontaneously terminate a connection due to internal or network problems.

2. Write down the short note on ATM (Asynchronous Transfer Mode). [WBUT 2018]

Answer:

ATM stands for Asynchronous Transfer Mode which is a switching technique used by telecommunication networks that uses asynchronous time-division multiplexing to encode data into small, fixed-sized cell. This is different from Ethernet or Internet, which use variable packet sizes for data or frames. ATM is the core protocol used over the synchronous optical network (SONET) backbone of the integrated digital service network (ISDN).

The ATM provides data link layer services that run on the OSI's layer 1 physical links. It operates on small a packet-switched and circuit-switched network which makes it deal for real time, low latency data such as VOIP (voice over internet protocol) and video.



3. a) What is the difference between the flow control and the congestion control? Justify for the long haul communication, the window flow control is ineffective.
b) Compare reservation based congestion control with permit based congestion control. **[MODEL QUESTION]**

Answer:

a) Flow control vs. congestion control:

Flow control means preventing the source from sending data that the sink will end up dropping because it runs out of buffer space. This is fairly easy with a sliding window protocol--just make sure the source's window is no larger than the free space in the sink's buffer. TCP does this by letting the sink advertise its free buffer space in the window field of the acks. Congestion control means preventing (or trying to prevent) the source from sending data that will end up getting dropped by a router because its queue is full. This is more complicated, because packets from different sources travelling different paths can converge on the same queue.

In long haul communication networks, bandwidth is usually so limited that calls for memory buffers or processor cycles be satisfied in much less time than it takes to send/receive a message. Credits given to senders can be honoured without necessarily tying up more than a small amount of buffers, typically one for each message being

reassembled. That is, receivers can "lie" in promising buffers which may not be available at the time credits are sent. Nevertheless, by the time buffers are needed, the receiver can manage to find some. A common strategy is to share a buffer pool among many receivers, or alternatively, to page out inactive buffers.

b) Reservation-Based –the hosts attempt to reserve network capacity when the flow is established.

-The routers allocate resources to satisfy reservations or the flow is rejected.

-The reservation can be receiver-based (e.g., RSVP) or sender-based.

- Permit-Based –The sender's rate is controlled by the receiver indicating the bits per second it can absorb.

4. What is congestion? Why do congestion occur? Explain Leaky bucket algorithm for congestion control. **[MODEL QUESTION]**

Answer:

The leaky bucket is an algorithm used in packet switched computer networks and telecommunications networks to check that data transmissions conform to defined limits on bandwidth and burstiness (a measure of the unevenness or variations in the traffic flow). The leaky bucket algorithm is also used in leaky bucket counters, e.g. to detect when the average or peak rate of random or stochastic events or stochastic processes exceed defined limits.

The Leaky Bucket Algorithm is based on an analogy of a bucket .that has a hole in the bottom through which any water it contains will leak away at a constant rate, until or unless it is empty. Water can be added intermittently, i.e. in bursts, but if too much is added at once, or it is added at too high an average rate, the water will exceed the capacity of the bucket, which will overflow.

There are actually two different methods of applying this analogy described in the literature .These give what appear to be two different algorithms, both of which are referred to as the leaky bucket algorithm. This has resulted in confusion about what the leaky bucket algorithm is and what its properties are.

In one version, the analogue of the bucket is a counter or variable, separate from the flow of traffic, and is used only to check that traffic conforms to the limits, i.e. the analogue of the water is brought to the bucket by the traffic and added to it so that the level of water in the bucket indicates conformance to the rate and burstiness limits. This version is referred to here as the leaky bucket as a meter. In the second version, the traffic passes through a queue that is the analogue of the bucket, i.e. the traffic is the analogue of the water passing through the bucket. This version is referred to here as the leaky bucket as a queue. The leaky bucket as a meter is equivalent to (a mirror image of) the token bucket algorithm, and given the same parameters will see the same traffic as conforming or nonconforming. The leaky bucket as a queue can be seen as a special case of the leaky bucket as a meter.

5. a) What is the difference between congestion control and flow control?
b) Explain the working of leaky bucket algorithm. Give argument why the leaky bucket should allow just one packet pet kick independent of how large the packet is. **[MODEL QUESTION]**

Answer:

a)

Congestion control	Flow Control
When there is a bursty situation over the network path and which packet should be chosen and how are the subject of this mechanism	The over all packet and data transmission over a network path is controlled by this mechanism.
Mainly Transport Layer responsibility.	Mainly Datalink layer responsibility though every layer governs the data flow in their own manner.
Token bucket, Leaky Bucket etc algorithms are used.	Stop and Wait, Selective Repeat ARQ, Routing algorithms are used.

b) The leaky bucket is an algorithm used in packet switched computer networks and telecommunications networks to check that data transmissions conform to defined limits on bandwidth and burstiness (a measure of the unevenness or variations in the traffic flow). The leaky bucket algorithm is also used in leaky bucket counters, e.g. to detect when the average or peak rate of random or stochastic events or stochastic processes exceed defined limits.

The Leaky Bucket Algorithm is based on an analogy of a bucket .that has a hole in the bottom through which any water it contains will leak away at a constant rate, until or unless it is empty. Water can be added intermittently, i.e. in bursts, but if too much is added at once, or it is added at too high an average rate, the water will exceed the capacity of the bucket, which will overflow.

There are actually two different methods of applying this analogy described in the literature .These give what appear to be two different algorithms, both of which are referred to as the leaky bucket algorithm. This has resulted in confusion about what the leaky bucket algorithm is and what its properties are.

In one version, the analogue of the bucket is a counter or variable, separate from the flow of traffic, and is used only to check that traffic conforms to the limits, i.e. the analogue of the water is brought to the bucket by the traffic and added to it so that the level of water in the bucket indicates conformance to the rate and burstiness limits. This version is referred to here as the leaky bucket as a meter. In the second version, the traffic passes through a queue that is the analogue of the bucket, i.e. the traffic is the analogue of the water passing through the bucket. This version is referred to here as the leaky bucket as a queue. The leaky bucket as a meter is equivalent to (a mirror image of) the token bucket algorithm, and given the same parameters will see the same traffic as conforming or nonconforming. The leaky bucket as a queue can be seen as a special case of the leaky bucket as a meter.