学　院
班　级
学　号
姓　名

密　封　线

东 北 大 学 考 试 试 卷（ 测 验 一 ）

2015　—2016　学年　第　一　学期

课程名称：计算机网络（Computer Networks）

总分	一	二	三	四	五	六	七	八

1. **Circuit switching versus packet switching. (10')**

a. Suppose that all of the network sources send data at a constant bit rate. Would packet-switching or circuit-switching be more desirable in this case? Why?

b. Suppose that all of the network sources are bursty—that they only occasionally have data to send. Would packet-switching or circuit switching be more desirable in this case? Why?

a. Circuit-switching is more desirable here because there are no statistical multiplexing gains to be had, and by using circuits, each connection will get a constant amount of bandwidth that matches its CBR rate. On the other hand, circuit-switching has more overhead in terms of signaling needed to set up the call, so there is an argument that packet-switching is preferable here since there is no call setup overhead with packet-switching. If this were an exam question either answer would be correct (as long as you provide the correct reasoning!).

b. Packet-switching is better here because there are statistical multiplexing gains—when a source does not have data to send, it will not be allocated bandwidth (it would be unused when the source had nothing to send). With packet-switching, this bandwidth is available for use by other sources.

2. **Circuit switching. (15')**

Consider sending a packet of F bits over a path of Q links. Each link transmits at *R* bps. The network is lightly loaded so that there are no queuing delays. Propagation delay is also negligible.

a. Suppose the network is a packet-switched datagram network and a connection-oriented service is used. Suppose each packet has h * F bits of header where 0 < h < 1. Assuming t_s setup time, how long does it take to send the packet?

b. Suppose that the network is a circuit-switched network. Furthermore, suppose that the transmission rate of the circuit between source and destination is *R*/24 bps. Assuming t_s setup time and no bits of header appended to the packet, how long does it take to send the packet?

c. When is the delay longer for packet switching than for circuit switching assuming h=0.5. Interpret your result?

a. The time required to transmit the packet over one link is (1 + h)F/R. The time required to transmit the packet over Q links is Q(1 + h)F/R Thus, the total delay for packet switching is $Q(1 + h)F/R+t_s$.

b. For circuit switching, bits are not "store and forwarded" before each link. Thus, there is only one transmission delay of F/(R/24) = 24F/R. The total delay is $24F/R+t_s$.

c. The delay is longer for packet switching when Q(1 + h)F/R + ts >24F/R + ts or equivalently when Q>16. Thus, if there are more than 16 links, packet switching has a larger delay, due to the store and forwarding. If there are fewer than 16 links, circuit switching has a larger delay, due to its reduced transmission rate.

学　院
班　级
学　号
姓　名

密 封 线

3. **Delays with multiple links. (15')**

Consider a packet of length L which begins at end system A, travels over one link to a packet switch, and travels from the packet switch over a second link to a destination end system. Let d_i, s_i and R_i denote the length, propagation speed, and transmission rate of link i, for i = 1, 2. The packet switch delays each packet by d_{proc}. Assuming no queuing delays, in terms of d_i, s_i, R_i, (i = 1, 2) and L, what is the total end-to-end delay for the packet? Suppose the packet is 1,000 bytes, the propagation speed on both links is $2.5 * 10^8$ m/s the transmission rates of both links is 1 Mbps, the packet switch processing delay is 1 msec, the length of the first link is 4,000 km, and the length of the last link is 1,000 km. For these values, what is the end-to-end delay?

The first end system requires L/R_1 to transmit the packet onto the first link;the packet propagates over the first link in d1/s1 the packet switch adds a processing delay of d_{proc} after receiving the entire packet, the packet switch requires L/R_2 to transmit the packet onto the second link; the packet propagates over the second link in d2/s2 Adding these five delays gives dend-end = L/R1 + L/R2 + d1/s1 + d2/s2 + dproc.

To answer the second question, we simply plug the values into the equation to get 8 + 8 + 16 + 4 + 1 = 37 msec.

4. **Store and forwarding. (10')**

In Question 3, suppose R_1=R_2=R and d_{proc} = 0.Furthermore, suppose the packet switch does not store-and-forward packets but instead immediately transmits each bit it receives before waiting for the packet to arrive. What is the end-to-end delay?

Because bits are immediately transmitted, the packet switch does not introduce any delay; in particular, it does not introduce a transmission delay. Thus, dend-end = L/R + d1/s1 + d2/s2 .

For the values in Question 3, we get . 8 + 16 + 4 = 28 msec.

5. **Protocol layers. (10')**

a. What are the five protocol layers, from top to bottom, in the Internet?

b. For each of the five layers, what is the name of the packets processed at the layer?

c. An end-system processes up to which layer?

d. A router processes up to which layer?

e. A link-layer switch processes up to which layer?

a. application, transport, network, link, physical

b. message, segment, datagram, frame, packet

c. an end-system processes up through the application layer

d. a router processes up through the network layer

e. a link-layer switch processes up through the link layer

6. **Carrier sense and collision detection. (10')**

Consider two nodes A and B on the same Ethernet segment, and suppose the propagation delay between the two nodes is 225 bit times. Suppose at time t=0 both nodes A and B begin to transmit a frame. At what time(in bit times) do they detect the collision? Assuming both nodes transmit a 48-bit jam signal after detecting a collision, at what time (in bit times) do nodes A and B sense an idle channel? How many seconds is this for a 10 Mbps Ethernet?

Both nodes A and B detect the collision at time t=225. At time t = 225 + 48 = 273 both nodes stop transmitting their jam signals. The last bit of the jam signal from B arrives at A at time t = 273 + 225 = 498 bit times. Similarly, the last bit of the jam signal from B arrives at A at time t = 273 + 225 = 498 bit times. For a 10 Mbps Ethernet, this corresponds to (498 bits)/(10^7 bits/sec) = 49.8 microseconds.

学　院
班　级
学　号
姓　名

密　封　线

7. **Link-layer services and Ethernet. (15')**

In the lecture we list a number of different services that a link layer can potentially provide to the network layer. These services include: a) framing, b) medium access, c) reliable delivery, d) flow control, e) error detection, f) error correction, g) full-duplex and half-duplex. For each of these services, discuss how or how not Ethernet provides the service.

a. Framing: Ethernet encapsulates the payload (such as an IP datagram) in an Ethernet frame. Included in this encapsulation is the preamble, which helps the receiving node determine where the frame begins and helps the receiving node synchronize its clock to the frame.

b. Ethernet provides CSMA/CD medium access.

c. Reliable delivery: Ethernet does not provide reliable delivery. Receivers do not send ACKS or NACKS to senders; senders do not maintain timers for transmitted frames. Thus, if a receiver determines that a frame has errors, it simply discards the frame. Higher-layer protocols may eventually retransmit the frame.

d. Ethernet does not provide flow control. Thus, if the network layer in the receiving node does not read data out of the adapter fast enough, the sender can overflow the link-layer receive buffer in the adapter.

e. Ethernet does perform error-detection using the CRC field in the Ethernet frame. If an error is detected, it discards the frame.

f. Ethernet does not correct bit errors.

g. Generally, CSMA/CD is half-duplex, as packets collide if transmitted at the same time. However, if all nodes are connected through a full-duplex switch, then Ethernet is full-duplex.

8. **Multiple access protocols: voice-over-IP and data. (15')**

In this chapter, we studied a number of multiple access protocols, including TDMA, CSMA, slotted Aloha, and token passing.

a. Suppose you were charged with putting together a large LAN to support IP telephony (only) and that multiple users may want to carry on a phone call at the same time. Recall that IP telephony digitizes and packetizes voice at a constant bit rate when a user is making an IP phone call. How well suited are these four protocols for this scenario? Provide a brief (one sentence) explanation of each answer.

b. Now suppose you were charged with putting together a LAN to support the occasional exchange of data between nodes (in this part of this question, there is no voice traffic). That is, any individual node does not have data to send very often. How well suited are these four protocols for this scenario? Provide a brief (one sentence) explanation of each answer.

c. Now suppose the LAN must support both voice and data and you must choose one of these multiple access strategies in order to support both applications on the same network, with the understanding that voice calls are more important than data. Which would you choose and why? How would voice and data be sent in this scenario? That is, which access protocol would you use, or adapt/modify, and why?

a. TDMA works well here since it provides a constant bit rate service of 1 slot per frame. CSMA will not work as work well here (unless the channel utilization is low) due to collisions and variable amount of time needed to access the channel (for example, channel access delays can be unbounded) and the need for voice packets to be played out synchronously and with low delay at the receiver. Slotted Aloha has the same answer as CSMA. Token passing works well here since each station gets a turn to transmit once per token round, yielding an essentially constant bit rate service.

b. TDMA would not work well here as if there is only one station with something to send, it can only send once per frame. Hence, the access delays are long (one half frame time on average), and the throughput over a long period of time is only 1/N of the channel capacity. CSMA would work well since at low utilization, a node will get to use the channel as soon as it need to. Slotted Aloha has the same answer as CSMA Token passing would work better than TDMA but slightly less well than CSMA and Slotted Aloha, since it must wait for the token to be passed to the other stations (who likely wouldn't use it) before sending again.

c. Here are two possible answers. One approach would be to divide the channel into two "pieces"-one for data packets and one for voice. This can be accomplished by assigning some number of TDMA slots for voice calls (for example, one slot to each user). Also, add some additional slots and allow the stations with data to send to perform random access (for example, slotted aloha or CSMA) within those data slots only. A second approach would be to use token passing with priorities, and give priority to voice packets.