Name_______________ Student ID____________________ Department/Year___________

Final Examination

Introduction to Computer Networks
Class#: 901 E31110
Fall 2018

10:20-12:10 Thursday
January 10, 2019

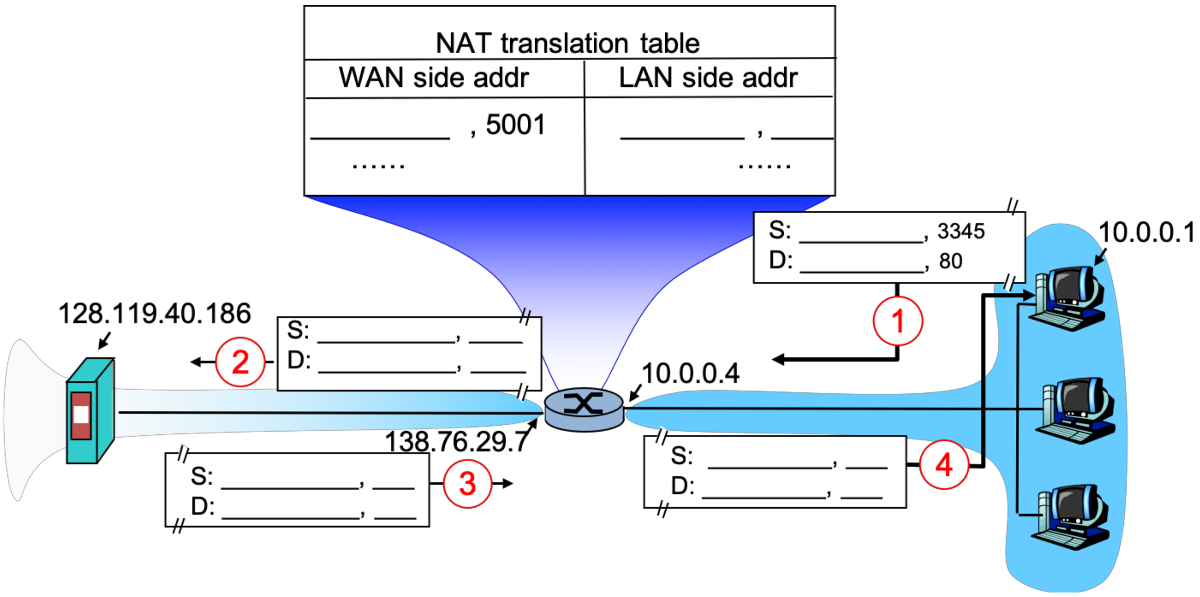
Prohibited

1. You are not allowed to write down the answers using pencils. Use only black- or blue-inked pens.
2. You are not allowed to read books or any references not on the question sheets.
3. You are not allowed to use calculators or electronic devices in any form.
4. You are not allowed to use extra sheets of papers.
5. You are not allowed to have any oral, visual, gesture exchange about the exam questions or answers during the exam.

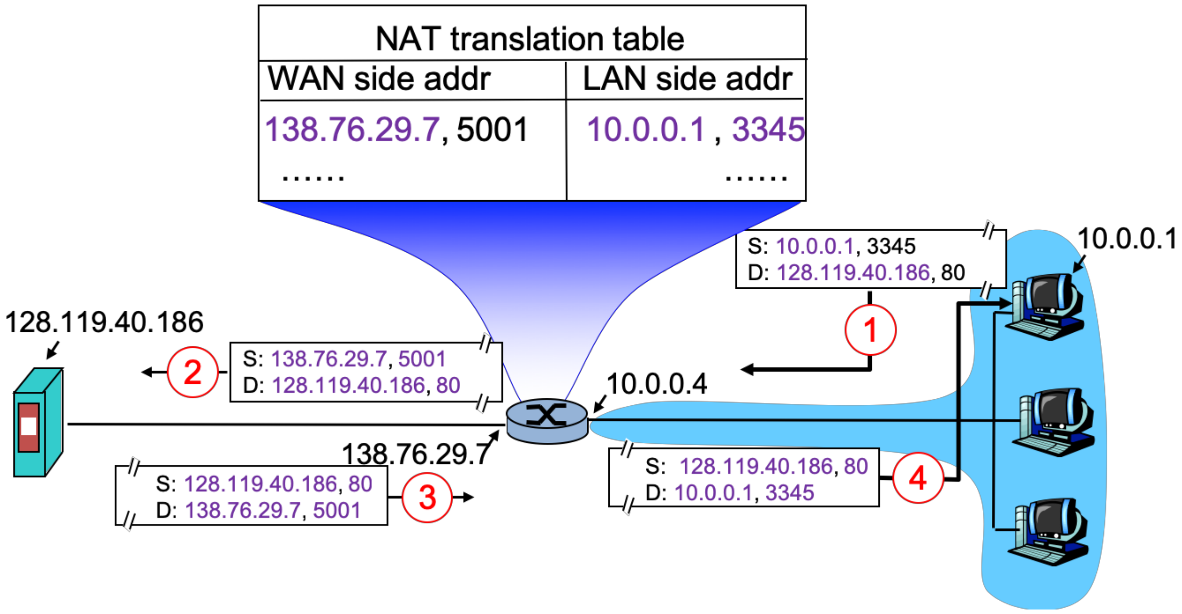
Cautions

1. Check if you get 14 pages (including this title page), 7 questions.
2. Write your **name in Chinese**, student ID, and department/year down on top of the first page.
3. There are in total 100 points to earn. You have 100 minutes to answer the questions. Skim through all questions and start from the questions you are more confident with.
4. Use only English to answer the questions. Misspelling and grammar errors will be tolerated, but you want to make sure with those errors your answers will still make sense.
5. If you have any extra-exam emergency or problem regarding the exam questions, raise your hand quietly. The exam administrator will approach you and deal with the problem.

1. Most homes are allowed only 1 IP address for Internet access. To allow multiple smart devices accessing the Internet simultaneously at home, a common practice is to run NAT at the home gateway. Below illustrates a home user at 10.0.0.1 requesting a Web page from a Web server at 128.119.40.186 via the home gateway. Based on your knowledge of how NAT operates, please fill in the blanks, including 2 phases of the outgoing packet, 2 phases of the returning packet, and the entry added to the NAT table. (10%)



Sample Solution:



2. Compare and contrast Go-Back-N (GBN) and Selective Repeat (SR).
   (a) Which method uses more network bandwidth to retransmit in case of losses? Why? (4%)
   (b) Which method requires more memory space at the receiver end? Why? (4%)
   (c) Which method requires more timers at the sender end? Why? (4%)
   (d) If the network bandwidth is abundant and memory is cheap which method would you choose to implement? Why? (4%)

Sample Solution:

(a) GBN. GBN always retransmits from the smallest, unack'd packet to the end of the batch (ack'd or not), whereas SR retransmits only the unack'd packet.
(b) SR. The packets received out of order need to be kept temporarily at the receiver buffer before the packets expected (in order) are received successfully.
(c) SR. GBN uses only one timer per batch of packets sent. SR uses one timer per packet sent.
(d) Multiple answers possible here. Take your pick and justify.

3. Below is the pseudo code that describes the reliable data transfer part of the TCP sender. The mechanism to adjust the window size for congestion control is missing. Please identify the code block where the missing functionalities shall be added and justify your pick.

```
     NextSeqNum = InitialSeqNum
     SendBase = InitialSeqNum
     CongestionWindowSize = InitialCongestionWindowSize

     loop (forever) {
         switch(event)
         event: data received from application above
             create TCP segment with sequence number NextSeqNum
   ┌         If (NextSeqNum < SendBase + CongestionWindowSize) {
   │             if (timer currently not running)
 A ┤                 start timer
   │             pass segment to IP
   └             NextSeqNum = NextSeqNum + length(data)
   ┌         } else
 B ┤
   └             refuse data
         event: timer timeout
   ┌         retransmit from the smallest, not-yet-acknowledged segment
 C ┤
   └         start timer
         event: ACK received, with ACK field value of y
   ┌         if (y > SendBase) {
   │             SendBase = y
 D ┤             if (there are currently not-yet-acknowledged segments)
   └                 (re)start timer
   ┌         } else {
   │             increment count of dup ACKs received for y
 E ┤             if (count of dup ACKs received for y = 3) {
   │                 resend segment with sequence number y
   └             }
     }   /* end of loop forever */
```

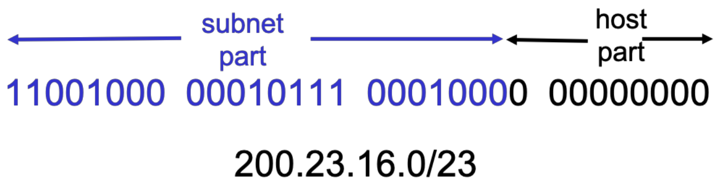

(a) Where to add the code to increase the congestion window size? Why? (4%)
(b) Where to add the code to reduce the congestion window size to 1? Why? (4%)
(c) Where to add the code to reduce the window size to half? Why? (4%)
(d) Where to add the code to set the slow start threshold to half of the window size? Why? (4%)

Sample Solution:

(a) Part D. It should be the part of the code that a fresh ack is received.
(b) Part C. It should be the part of the code that a timeout event occurs.
(c) Part E. It should be the part of the code that 3 duplicate acks are detected.
(d) Part C and E. It should be the part of the code that a packet loss is detected, whether it is due to timeout or 3 duplicate acks.

4. SouthPole is an ISP which owns the IP address block: 200.23.16.0/23. A known tradition at this ISP is to allocate IP addresses consecutively from the smallest number available. Ever since the penguins went extinct, business has been quiet at SouthPole. This year, Santa and Elf decide to move to the south pole for a change and subscribe to SouthPole.
   (a) What is the smallest and the largest IP address that can be offered by SouthPole? (4%)
   (b) Suppose Santa requests first, for a block of 32 IP addresses. What would be the smallest and the largest IP address assigned to Santa? Write the block of addresses in x.x.x.x/x format. (4%)
   (c) Continue from (b). Next, Elf requests for a block of 64 IP addresses. What would be the smallest and the largest IP address assigned to Elf? Write the block of addresses in x.x.x.x/x format. (4%)
   (d) Suppose Santa and Elf request for the 32 and 64 IP addresses simultaneously. What address blocks would the ISP assign such that only one x.x.x.x/x is necessary for Santa and Elf each? (4%)

Sample Solution:



(a) Smallest : 11001000 00010111 00010000 00000000 (200.23.16.0)
    Largest : 11001000 00010111 00010001 11111111 (200.23.17.255)

(b) A block of 32 IP addresses, consecutively assigned from the smallest:
    From : 11001000 00010111 00010000 00000000 (200.23.16.0)
    To : 11001000 00010111 00010000 00011111 (200.23.16.31)
    In x.x.x.x/x format: 200.23.16.0/27

(c) A block of 64 IP addresses, consecutively assigned from the smallest:
    From : 11001000 00010111 00010000 00100000 (200.23.16.32)
    ~ : 11001000 00010111 00010000 00111111 (200.23.16.63)
    ~ : 11001000 00010111 00010000 01000000 (200.23.16.64)
    To : 11001000 00010111 00010000 01011111 (200.23.16.95)

In x.x.x.x/x format: 200.23.16.32/27 and 200.23.16.64/27

(d) Assign for Elf's block of 64 first and then Santa's block of 32:

B: 64

From : 11001000 00010111 00010000 00000000 (200.23.16.0)

To : 11001000 00010111 00010000 00111111 (200.23.16.63)

In x.x.x.x/x format: 200.23.16.0/26

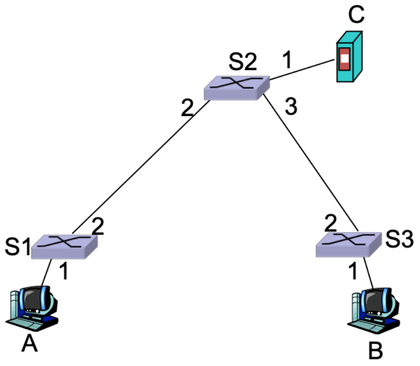
A:32

From : 11001000 00010111 00010000 01000000 (200.23.16.64)

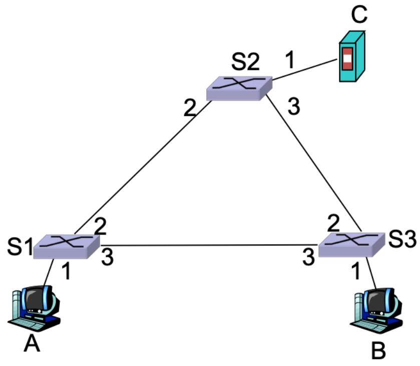
To : 11001000 00010111 00010000 01011111 (200.23.16.95)

In x.x.x.x/x format: 200.23.16.64/27

5. Network A and Network B are two brand-new institutional networks connected by 3 link-layer switches, namely S1, S2, and S3. Link S1-S2, S2-S3, and S3-S1 have the same link bandwidth and the same length. A user walks to computer A and sends an echo request (ping message) to server C once.



Network A



Network B

(a) Consider Network A. Show the switch table in S1, S2, and S3 respectively, after the user receives an echo reply (ping reply message) from server C. (4%)

(b) Now, consider Network B. Show the switch table in S2, and S3 respectively, after the echo request from computer A reaches S2 and S3 for the first time. If a packet destined for computer A arrives at S2 right at this moment, where would S2 forward the packet? (3%)

(c) Continue from (b). Show the switch table in S2, and S3 respectively, after the echo request reaches S2 and S3 for the second time. If a packet destined for computer A arrives at S2 right at this moment, where would S2 forward the packet? (3%)

(d) Do you see any problem using self-learning algorithm in Network B? How would you advise Network B's administrator to avoid the problem? (4%)

Sample Solution:

(a) S1: (A,1)(C,2)
S2: (A,2)(C,1)
S3: (A,2)

(b) S2: (A,2), S3: (A,3)
To interface 2, or S2->S1

(c) S2: (A,3), S3: (A,2)
To interface 3, or S2->S3

(d) Any problem that makes sense will be accepted. Here are a couple problems people may see:

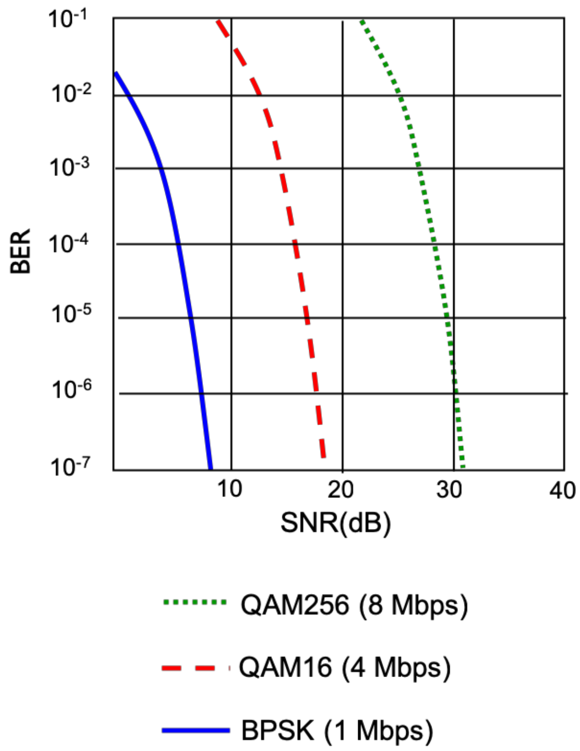
* Forwarding of the echo request message will never end.
* Packets may loop around the network indefinitely.
* Packets may travel a longer path

Any suggestions that makes sense will be accepted. Here are a couple possible suggestions people provide:

* Avoid having cycles in the network connected by link-layer switches
* Use network-layer routers to connect the subnets if cycles are not avoidable
* Adding a sequence # or counting the time the messages hop around

6. QAM256, QAM16, and BPSK are three mechanisms to modulate bits onto the electromagnetic waves. In QAM-based modulation mechanisms, the transmitter varies the amplitude and phase to create distinct states of a radio wave. In QAM16 for example, 4 levels of amplitude and 4 levels of phase shift allow 16 distinct states, therefore, allowing modulation of 4 bits. Similarly, using QAM256, one can modulate 8 bits of data per radio wave. BPSK is short for binary phase shift keying, which uses only 2 levels of phase shift and therefore modulates only 1 bit of data per radio wave.

   Although QAM256 allows a higher bit rate than the other two mechanisms, it is more vulnerable to noise and interference, as it is harder to differentiate amplitude and phase correctly at a finer granulation. This is why in the figure below one sees how the bit error rate (BER) of QAM256 is much higher than QAM16 and BPSK given the same level of signal to Noise ratio (SNR).



   (a) If one wishes to keep the BER below 10^{-3} over a wireless link, which modulation mechanism should the link layer protocol use for SNR at 30dB, 20dB, and 10dB respectively? (6%)
   (b) Provided the background above, argue why WiFi transmission rate drops as one moves away from the WiFi AP. (4%)

Sample Solution:

Long question but easy to answer

(a) 30dB: Any of QAM256, QAM16, or BPSK
    20dB: Either QAM16 or BPSK
    10dB: Only BPSK

(b) When the receiver walks away from the transmitter, the radio signal strength drops, and therefore the SNR decreases. To maintain the BER, WiFi adapts the transmission rate by changing the modulation mechanism to the ones that are less vulnerable to noise and interference.

    Or, when the lowest-rate modulation mechanism is already in use, walking further away results in an even lower level of SNR. The BER will go higher than the level that was intended and the rate of successfully transmitting packets will drop further and in effect prolonging the time needed to complete the data transmission.

7. Transmit this data sequence D=100010 over a link using CRC.
   (a) Suppose the CRC uses a G=111, what will the EDC bits be? (3%)
   (b) Continue from (a). If the 2nd, 3rd, and 4th bits of the D+EDC sequence are flipped as the sequence is transmitted through the link, will the receiver be able to detect the error? (3%)
   (c) Suppose the CRC uses a G=1111, what will the EDC bits be? (3%)
   (d) Continue from (c). If the 2nd, 3rd, and 4th bits of the D+EDC sequence are flipped as the sequence is transmitted through the link, will the receiver be able to detect the error? (3%)
   (e) Continue from (c). If the 2nd, 3rd, 4th, and 5th bits of the D+EDC sequence are flipped as the sequence is transmitted through the link, will the receiver be able to detect the error? (3%)
   (f) It is mentioned in the lecture – CRC using an r-bit G is capable of detecting all burst errors less than r bits. What about the other way around – is CRC using an r-bit G capable of detecting all burst errors of r bits or above? Argue for your answer. (3%)

Sample Solution:

(a) 11

```
       110101                  110101
   111√10001000            111√10001011
       111                     111
        110                     110
        111                     111
          110                     110
          111                     111
            100                     111
            111                     111
             11                       0 ← perfectly divided
```

(b) No

```
      100101
111√11111011
    111
      110
      111
        111
        111
          0 ← perfectly divided, CRC returns 'no error detected'
```

(c) 000

```
        110000                    110000
1111√100010000            1111√100010000
     1111                      1111
       1111                      1111
       1111                      1111
          00000                     00000 ← perfectly divided
```

(d) Yes

```
        100011
1111√111110000
     1111
        01000
        1111
          1110
          1111
            001 ← not perfectly divided, CRC returns 'error detected'
```

(e) No

```
        100011
1111√111100000
     1111
        000000 ← perfectly divided, CRC returns 'no error detected'
```

(f) No, at least for r=3 or 4. We see two examples in (b)(e) that a 3-bit G=111 can't detect a 3-bit burst error and a 4-bit G=1111 can't detect a 4-bit burst error. Other arguments that are fuzzy will be partially accepted.