**American University of Beirut**

Department of Electrical and Computer Engineering

EECE 350 – Computer Networks

Spring 2015

**HOMEWORK 1 Solution**

**Problem 1 [15 points]**

Let XY be the last two digits of your ID number. Find the RFC with number 6601+XY (i.e., if your ID number is 2014 12345, the RFC number is 6601 + 45 = 6646. If your ID number ends in 98, use RFC 6601). Answer the following questions:

Assuming XY=45.

The RFC Number is 6601+45=6646.

1. [5] What is the title of the RFC?

“DECoupled Application Data Enroute (DECADE) Problem Statement”

1. [5] Who is (are) the author(s) of the RFC?

H. Song, N. Zong, Y. Yang, R. Alimi

1. [5] When was the RFC published?

July 2012

### Problem 2 [35 pts]

Consider the Roman imperial postal service, in service 2 millennia ago. Imperial messages were communicated through couriers who traveled from one station to the other, relaying written and sometimes encrypted messages to fresh couriers at each. Each courier in turn rode towards the next station along the way to the destination, and brought back the reply to be delivered back to Rome. This system was used to communicate quickly with far away provinces, sometimes through hostile territory.  
  
[5 points per layer] Compare this system to the 7-layer ISO OSI architecture, providing the equivalent of each OSI layer in the Roman network. One sentence per layer!

To start at the bottom layer, Romans built paved roads across all the Roman Empire to speed up communications between Rome and the provinces. These are the equivalents of the physical-layer links of today’s networks.

Network nodes correspond to the mail relays where a message would get exchanged between couriers.

The equivalent of packets were hand written messages carried by the couriers. The data link layer functions were performed through the use of couriers galloping or walking between two mail stops.

The network layer functionalities of end-to-end addressing and delivery correspond to the addressing of the message and the knowledge of each mail stop’s staff as to how to get to the next relay towards the destination, or through specific instructions by the sender. Couriers on the intermediate road segments usually did not need to worry about the last delivery step, and local delivery was done by people who were familiar with the area, similarly to how routing is performed in the Internet.

The end-to-end reliable communication (of a modern transport layer) was achieved through recipients always writing back when requested to do so, and therefore the sender would know when to expect a response back. The sender would re-send a letter when the expected response is not received.

It is not straightforward to find an equivalent to a session layer in the postal system given the application of interest. However, it is clear that presentation layer functionality was needed when encrypted messages were exchanged.

The application layer corresponds to writing/reading letters.

### Problem 3 [20 pts]

Suppose users share a 1 Mbps link. Also suppose each user requires 100 kbps when transmitting, but each user transmits only ***X*** percent of the time, where ***X*** = 10 + (last two digits of your ID number)/20. For example, if your ID number is 2014 12345, ***X*** = 10 + 45/20 = 12.25.

Assuming X= 10 + 45/20 = 12.25

1. [5] When circuit switching is used, how many users can be supported?

When circuit switching is used, each user is allocated 100 kbps all the time (irrespective of probability of transmission.) Therefore 1000 kbps / 100 kbps = 10 users max can be supported.

1. [5] For the remainder of this problem, suppose packet switching is used. Find the probability that a given user is transmitting.

This probability is *12.25*% (or *12.25*/100).

1. [5] Suppose there are 36 users. Find the probability that at any given time, exactly *n* users are transmitting simultaneously. (Hint: use the binomial distribution.)

Let *x* = *12.25*/100.

Probability =

Probability =

1. [5] Find the probability that there are 11 or more users transmitting simultaneously. How many users can be supported when packet switching is used?

The probability that 11 or more users are transmitting is the sum for all *n* between 11 and 36 (included). This is also equal to:

Packet switching can support a large number of users; but as the number of users increases, the probability that more than 10 are using the link increases, which causes queuing and excessive delay (up to packet drop) over link.

### Problem 4 [30 points]

Find the time at which a receiver completely receives a 1,000,000-Byte file in the following cases, assuming a symmetrical round-trip-time (RTT) of 50 msec from sender to receiver and back, and a packet size of ***X*** Bytes to which a header of 64 Bytes is added. Right after *t* = 0, assume that one RTT is needed for signaling between sender and receiver before data transfer can begin. The link bitrate is 5 Mbps.   
***X*** = 1500 + (last two digits of your ID number)×5.   
For example, if your ID number is 2014 12345, ***X*** = 1500 + 45×5 = 1725.

Assuming X=1500+ 40×5 = 1700 Bytes

1. [10] Data packets can be sent continuously back-to-back.

File size = *S* Bytes = 1,000,000 Bytes

The file is sent using packets = packets.

Each packet is *X*+64 Bytes = 1700 + 64= 1764 bytes, except the last one.

Total time = initial setup RTT + final propagation time (= RTT/2) + transmission time of all packets (headers are 64×*N* Bytes, and total data is *S* Bytes)

= 1.5×RTT + (8×64×*N* + 8×*S*)/5×106

= 1.7353136 s

1. [10] A data packet can only be sent after receiving a response for the previous one (assume the response is very small and therefore its transmission time is negligible).

Total time = initial RTT + final propagation time + transmission time of all packets (headers and data) + RTT after each packet except last one

= 1.5×RTT + (8×64×*N* + 8×*S*)/5×106 + (*N* – 1) × RTT

=

= 31.1353136 s

1. [10] Up to **10** packets can be sent per RTT.

We can send 10 packets per RTT.

To send all packets, we need to find *M* = = 58

During *M*×RTT, we send 10×*M = 580* packets.

We will be left with *S’* = (*S* – 10×*M*×*X*) = = 14000Bytes to send.

For the *S’* Bytes, we send them using = 9 packets.

Each of these *N’* packets is *1700*+64 = 1764 Bytes, except the last one.

Total time = initial RTT + final propagation time + *M*×RTT + transmission time of remaining *N’* packets (headers are 64×*N’* Bytes, and data is *S’* Bytes)

= 1.5×RTT + *M*×RTT + (8×64×*N’* + 8×*S’*)/5×106

= = 29.0983216 s

Assuming X=1500+ 45×5 = 1725 Bytes

The file is sent using packets = packets.

To send all packets, we need to find *M* = = 58

Total time = initial RTT + *M*×RTT = *59*×RTT = 2.95 ms