Homework1 Solutions

1.

a. Service time is 1 second. 2 points.

b. It takes 10 seconds to service every car, (10 cars * 1 seconds per car). 3 points.

c. It takes 25 seconds to travel to the next toll booth (500 km / 20 km/s). 5 points.

d. Just like in the previous question, it takes 25 seconds, regardless of the car. 5 points.

e. It takes 34 seconds until the first car gets serviced at the next toll booth (10-1 cars * 1 seconds per car + 500 km / 20 km/s). 5 points.

f. No, because cars can't get service at the next tollbooth until all cars have arrived. 5 points.

g. Yes, one notable example is when the last car in the caravan is serviced but is still travelling to the next toll booth; all other cars have to wait until it arrives, thus no cars are being serviced. 5 points.

2.

a. Yes, in practice, queuing delay can vary significantly. We use the above formulas as a way to give a rough estimate, but in a real-life scenario it is much more complicated. 5 points.

b. Queuing Delay = I(L/R)(1 - I) * 1000 = 0.1071*(4800/1300000)*(1-0.1071) * 1000 = 0.3531 ms. 5 points.

c. Queuing Delay = I(L/R)(1 - I) * 1000 = 0.2215*(4800/1300000)*(1-0.2215) * 1000 = 0.6367 ms. 5 points.

d. Packets left in buffer =packets-rounded down(time/delay)= a - floor(1000/delay) = 924 - floor(1000/0.6367) = 0 packets.

Floor value returns the closest integer less than or equal to a given number. 7.5 points.

e. Packets dropped = packets - buffer size = 924 - 744 = 180 dropped packets. 7.5 points.

3.

When sending a packet from a source host to a destination host over a fixed route in a computer network, the end-to-end delay consists of several components. These delay components can be categorized into four main types: transmission delay, propagation delay, processing delay, and queuing delay. Some of these delays are constant, while others are variable. Let's break down these delay components:
Transmission Delay: The time it takes to push all the bits of the packet into the link or transmission medium. This delay is constant for a given packet size and link bandwidth. It is determined by the physical properties of the link. 5 points.
Propagation Delay: The time it takes for a signal or packet to travel from the source to the destination. It is related to the distance between the two hosts and the propagation speed of the medium. This delay is constant as long as the distance and propagation speed remain unchanged. However, it can vary for different routes or network segments. 5 points.

Processing Delay: The time spent by networking devices (routers, switches, and hosts) to process the packet. This includes the time spent in routing, error checking, and other processing tasks. This delay is variable depending on the processing workload of the devices along the route. It can vary due to changes in device congestion and routing decisions. 5 points.
Queuing Delay: The delay that occurs when a packet is waiting in a queue to be transmitted. This typically occurs in routers or switches when the outgoing link is busy. This delay is highly variable and dependent on network congestion. During periods of high traffic, queuing delays can increase significantly. 5 points.

4. The Internet protocol stack, often referred to as the TCP/IP stack, consists of five layers. Let's explore each layer's principal responsibilities and explain encapsulation at each step using a top-down approach:

1. Application Layer: The Application Layer is the topmost layer and is responsible for providing network services directly to end-users and applications. It includes various protocols that facilitate communication between software applications. At this layer, data from an application is prepared for transmission over the network. It is encapsulated within application-specific data structures or messages. For example, in the case of web browsing, an HTTP request is created by the web browser, encapsulating the user's request for a web page. 4 points

2. Transport Layer: The Transport Layer is responsible for end-to-end communication between devices. It ensures data is reliably delivered, in order, and provides error checking and flow control. The data received from the Application Layer is divided into segments. Each segment includes a header containing information such as source and destination port numbers. This process is encapsulation at the transport layer. Examples of protocols operating at this layer include TCP and UDP. 4 points

3. Network Layer: The Network Layer focuses on routing packets of data between different networks. It deals with logical addressing, routing, and forwarding. The segments from the Transport Layer are further encapsulated into packets or datagrams. These packets include information like source and destination IP addresses. Routers use this information to determine the best path for packet delivery. IP (Internet Protocol) operates at this layer. 4 points

4. Data Link Layer: The Data Link Layer is responsible for the reliable transmission of frames between devices on the same network segment. It ensures data integrity and manages access to the physical medium. The packets from the Network Layer are encapsulated into frames. Frames include control information like MAC addresses for source and destination devices. Ethernet is a common Data Link Layer technology. 4 points

5. Physical Layer: The Physical Layer deals with the actual transmission of raw bits over the physical medium, considering electrical, mechanical, and functional characteristics of hardware. At the Physical Layer, the frames are converted into electrical signals, optical signals, or radio waves suitable for transmission over the physical medium. This involves encoding and modulation to convert digital data into a physical signal that can traverse the medium. 4 points

Encapsulation ensures that data is packaged appropriately at each layer of the protocol stack. As data flows from the Application Layer down to the Physical Layer, each layer adds its own header and possibly a trailer to the data, creating a hierarchical structure of encapsulated information. When data is received at the destination, the process is reversed, and each layer strips off its respective header and trailer until the original data is extracted and delivered to the receiving application. This encapsulation and decapsulation process allows for the proper transmission and reception of data across a network while maintaining the necessary control and addressing information at each layer.