1. F, T, F, F, T, F, T, T, T, F, T, F, F, T, F, T, T, F, F, T, T

2. 60Km, 14.3%, 820Hz, 01011111110111101010, 20Ksec, 2Gbps, 160bits, 1600bps, 10 frames/sec, 320 slots/sec, 1013.9bps, 4Mbps, 7bits, 2(x+y), 2.84sec, 2.72sec, 2.64sec, 2.76sec, 1.24sec, 1.25sec, 4-Sockets, 9-Sockets

3.
a. 4 FCS bits, 1011, Transmitted pattern : 10011011011. The red bits are the FCS bits
b. Received Pattern 01000101110. Eight errors did occur (Since the received Pattern is NOT the same as the Transmitted Pattern). Receiver will divide this pattern by the generator pattern and observe a zero remainder. The receiver was **NOT** able to detect the error. According to the receiver, the transmitted sequence was indeed 01000101110 and hence he will decode the message sequence as 0100010. Of course, he is wrong.

c. Received pattern = 10011011011 $\oplus$ 01000101000 = 11011110011. When we divide this pattern by the generator pattern, the reminder is 0110 which is **NOT** 0 and hence the receiver is able to detect the error (The receiver does not know how many errors or where are they located). Note the receiver does **NOT** know the error pattern. He only observes the received pattern**.**

4.

Throughput = 5000/15 = 333.3 bps

Link Utilization = 5/15 = 33.3%

Time	Action @ Transmitter	Action @ Receiver	Time
0	F_0 is transmitted		0
1	F_1 is transmitted		1
2	Sender window is closed	F_0 is received (No errors), ACK_0 is returned	2
3	ACK_0 is received. Window slides by 1 and F_2 is transmitted	F_1 is received and detected to be in error, Frame F_1 is dropped	3
4	Sender window is closed		4
5	F_1 is timed out and is retransmitted.	F_2 is received and ACK_2 is returned	5
6	ACK_2 is received. Window does NOT slide		6
7	Sender window is still closed	F_1 is received and ACK_1 is returned	7
8	ACK_1 is received. Sender window slides by two units. F_3 is transmitted. This frame is lost in channel		8
9	$F_4 = F_0$ (the new one) is transmitted		9
10	Sender window is closed		10
11	Sender is idle	F_4 is received and ACK_4 is returned	11
12	F_3 is timed out and is retransmitted. ACK_4 is received		12
13	Sender is idle		13
14	Sender is idle	F_3 is received and ACK_3 is returned	14
15	ACK_3 is received. Sender is done		15

5.

a.

Step	Action	Delay (sec)
1	Client A request the IP address of X from local DNS server	0.1
2	Local DNS server contact the RNS for IP address of X. The RNS server return the IP address of the TLD.	0.8 (Round Trip)
3	Local DNS server contact the TLD for IP address of X. The TLD server is assumed to be the Authoritative name server of X	0.6 (Round Trip)
5	The Local DNS server returns the IP address of X to the client DNS which in turn passes it to HTTP	0.1
6	Client A requests setting up a TCP connection with the server X (Handshaking)	0.4 (Round Trip)
7	Client A request downloading the HTML page	0.2
8	Page downloaded	(1G/1G) + (1G/1M) + (1G/1G) + 0.2 = 1002.2
9	**Total Delay**	**1004.4 sec**

b. It will take host B only **200 msec** to get the IP address of X which is cached in the local DNS server

c. Average rate = 0.5(1Gbps) + 0.5(1Mbps) = **500.5Mbps**. Note that the 1 Mbps constitute the "bottleneck" of the path between A and X.