ITS 323 – QUIZ 3 ANSWERS

First name: ______________________________ Last name: ___________________________

ID: _____________________________ Total Marks: ____________
out of 10

In all questions, assume bits are number left to right. That is, for the sequence 010111, the first bit is 0, the second bit is 1, the third bit is 0 and the sixth bit is 1.

Question 1 [2 marks]

Consider a simplified CRC error detection algorithm where there are *k* bits of data to send, the frame check sequence (which is appended to the end of the data) is *f* bits in length, and the divisor must be (*f* + 1) bits in length. If the data to send is 1010 and the divisor is 110:

a) What is the value (in binary) of the frame check sequence? [1 mark]

b) If the 3rd, 4th and 5th bits are received in error, can the receiver detect the errors? Show your calculations or explanation. [1 mark]

Answer

a. The frame check sequence (FCS) must be two bits in length (since divisor is 3 bits). Therefore the transmitted data is: 1010xx where xx = FCS = 10. This results in transmitted data of 101010 = 42, which is divisible by the divisor (110 = 6).

b. If the 3rd, 4th and 5th bits are in error, then 100100 = 36 is received. Since 36 is divisible by 6, then NO error is detected.

Question 2 [4 marks]

a) Assume a Hamming distance based forward error correction algorithm is used on a data block of 32 bits, and produces 50-bit codewords. If you instead increase the codeword size to 40 bits, in general, less errors can be detected.

True / False

b) Choosing a very long time-out interval for an ARQ protocol may lead to low throughput because a lot of time may be spend waiting for an ACK (in the case that the DATA frame was lost).

True / False

c) Using Pulse Code Modulation to encode analog data, according to the sampling theorem, a data rate of 6Mb/s is required. How many bits are used to represent each sample (code), if the highest frequency component of the analog data is 1MHz?

a. 1 bit
b. 2 bits
c. 3 bits
d. 4 bits
e. 5 bits
f. 6 bits
g. 12 bits

d) If a protocol uses an 6-bit field in the header for sequence numbers of frames (and all frames are the same size), according to the sliding window mechanism, the minimum number of frames a receiver should be able to store in its receive buffer is:

a. 0 frames (no buffer needed)

b. 1 frame

c. 5 frames

d. 6 frames

e. 31 frames

f. 32 frames

g. 63 frames

h. 64 frames

Answer

True. A larger codeword (relative to the data) means there is a greater chance that, if the codeword has errors, then it will be different from one of the valid codewords. A received codeword different from a valid codeword indicates an error.

True. With a long timeout, the source spends a lot of time waiting (not sending) if a DATA frame is lost, hence resulting in low throughput.

3 bits. The sampling theorem says you should sample at at least twice the rate of the highest frequency component (2 x 1Mhz = 2000000 samples per second). Each sample contains a single code, which contains 3 bits. Data rate will 6Mb/s.

63 frames. The maximum window size is 2^k-1 where k is the number of bits in the sequence number. The receiver must be able to receive the maximum window full of frames before sending an ACK, hence needs a buffer size to store at least 2^k-1 frames.

Question 3 [3 marks]

What is the maximum throughput of the Stop and Wait Flow Control protocol.

You can assume:

- Data rate is 2Gb/s
- Data frame size is 9000 bits of data plus 1000 bits of header
- ACK size is 1000bits
- Propagation time is 1.75μsec
- Processing delay is 0

Answer:

Total time for transmission of data is: DataTransmission + Propagation + Processing + AckTransmission + Propagation

DataTransmission: 10000 / 2Gb/s = 5us

AckTransmission = 0.5usms

Propagation = 1.2ms

Processing = 0ms

Throughput = 9000 bits / (5 + 1.75 + 0.5 + 0 + 1.75) us

≈ 9000 / 9 us

= 1000Mb/s = 1Gb/s

Question 4 [1 mark]

Go-Back-N ARQ with a *k* bit sequence number limits the maximum window size to 2^k-1. Explain a problem that may occur if the maximum window size was *greater than* 2^k (e.g. 2^k+1). (A diagram may help with your explanation).

Answer:

With a *k* bit sequence number, the range of numbers are: 0, 1, 2, …, 2^k-2, 2^k-1, 0, 1, …

If the window is larger than 2^k, then the window may encompass frames with the same sequence number (say, *i*). The problem with this is that when an ACK is returned with number (*i*+1), the source cannot be certain which frame the ACK acknowledges: the first frame with sequence number *i* or the second frame with the sequence number *i*?

Example: *k* = 2, window size is 5.

Sequence numbers:	0	1	2	3	0	1	2	3	0	…
Possible window		1	2	3	0	1				

If an ACK with number 2 is returned, does it acknowledge the first frame with sequence number 1 or the second frame with sequence number 1? This ambiguity should be not be present in a protocol.