BLG 337E- PRINCIPLES OF COMPUTER COMMUNICATIONS- FALL 2014

HOMEWORKS 1 & 2

Deadline: December 07, 2014 – 23:00
Submission: Upload your executable program files in ZIP format, through NINOVA.
No need to write a report.

1) Write a program that implements message flow from the top layer to the bottom layer of the 7-layer protocol model. Your program should include a separate protocol function for each layer. Protocol headers are sequence up to 64 characters. Each protocol function has two parameters: a message passed from the higher layer protocol (a char buffer) and the size of the message. This function attaches its header in front of the message, prints the new message on the standard output, and then invokes the protocol function of the lower-layer protocol. Program input is an application message (a sequence of 80 characters or less).
---

2) Write a program to implement CDMA. Assume that the length of a chip sequence is eight and the number of stations transmitting is four. Your program consists of three sets of processes: four transmitter processes (t0, t1, t2, and t3), one joiner process, and four receiver processes (r0, r1, r2, and r3). The main program, which also acts as the joiner process first reads four chip sequences from the standard input and a sequence of 4 bits (1 bit per transmitter process to be transmitted), and forks off four pairs of transmitter and receiver processes. Each pair of transmitter/receiver processes (t0,r0; t1,r1; t2,r2; t3,r3) is assigned one chip sequence and each transmitter process is assigned 1 bit (first bit to t0, second bit to t1, and so on). Next, each transmitter process computes the signal to be transmitted (a sequence of 8 bits) and sends it to the joiner process. After receiving signals from all four transmitter processes, the joiner process combines the signals and sends the combined signal to the four receiver processes. Each receiver process then computes the bit it has received and prints it to standard output. Use pipes for communication between processes.
---

3) Write a program to simulate the behavior of the CSMA/CD protocol over Ethernet when there are N stations ready to transmit while a frame is being transmitted. Your program should report the times when each station successfully starts sending its frame. Assume that a clock tick occurs once every slot time (51.2 μsec) and a collision detection and sending of a jamming sequence takes one slot time. All frames are the maximum length allowed.

---
4) You are to implement an error-detection mechanism using the standard CRC algorithm. Write two programs, *generator* and *verifier*. The *generator* program reads from standard input a line of ASCII text containing an n-bit message consisting of a string of 0s and 1s. The second line is the k-bit polynomial, also in ASCII. It outputs to standard output a line of ASCII text with n+k 0s and 1s representing the message to be transmitted. Then it outputs the polynomial, just as it reads it in. The *verifier* program reads in the output of the generator program and outputs a message indicating whether it is correct or not. Finally, write a program, *alter*, that inverts 1 bit on the first line depending on its argument, (the bit number counting the leftmost bit as 1), but copies the rest of the two lines correctly.
By typing “generator <file | verifier”, you should see that the message is correct,
but by typing “generator <file | alter arg | verifier”, you should get the error message.

*Resource: **“Computer Networks”**, 5th Edition, A. Tanenbaum and D.Wetherall, Prentice Hall, Pearson-2011*