

Department of Computer Science and Engineering
CSN-361: Computer Networks Laboratory (Spring 2020-2021)

Instructions to Students:

1. Join MS Team site for the course CSN-361. Team code will be shared over email.
 2. There are SIX problem statements, for which you need to write the program and test your code on your own PC. You should complete all SIX problems.
 3. For any problem statement, you can write your programs using **C++ or Java**.
 4. Create a folder named as your **EnrollmentNo**. Keep all the implemented codes (Problem Statement wise) within separate folders namely P1, P2, P3, etc. maintained within the parent folder namely **EnrollmentNo**.
 5. Prepare a report (PDF file) including the theory, implementation details and simulation results (with screenshots of successful runs of the code for testcases). Keep that report PDF file within the parent folder namely **EnrollmentNo**. Report template for LATEX: <https://www.overleaf.com/read/hcyvtxhryjzp>
 6. **Submit your** zipped folder (<EnrollmentNo>.zip or <EnrollmentNo>.tar.gz) through **MS Team** only under **Assignments** tab namely "Report and Code Folder Submission" before the strict deadline **May 5, 2021 11:59 PM IST**.
-

Problem Statement 1:

Write a C++ program to print the MAC address of your computer, the host name and the IP address of your computer.

Problem Statement 2:

Write a socket program in Java for PING command.

Problem Statement 3:

Implement an error detection mechanism using the standard CRC algorithm. Write two programs: generator and verifier. The generator program reads from standard input an n-bit message as a string of 0's and 1's as a line of ASCII text. The second line is the k-bit polynomial, also in ASCII. It first checks that the polynomial is not divisible by x and x+1. If it is divisible by x or x+1, it outputs error else it outputs to standard output a line of ASCII text with n+k-1 0's and 1's representing the message to be transmitted. Then it outputs the polynomial, just as it read it in. The verifier program reads in the output of the generator program (if output is not error) and outputs a message indicating whether it is correct or not. Finally write a program, alter, that inverts one bit in the first line of the output of the generator depending on its argument, but copies rest of the first line and second line correctly. Now type the following and report the outcome.

- (i) generator < file | verifier
 - (ii) generator < file | alter arg | verifier
-

Problem Statement 5:

Write a program that an instructor can use to demonstrate the method of calculating IPv4 checksum. Your program should ask the user to enter the values of different fields of an IPv4 header. It should then calculate IPv4 checksum. Your program should not only show the final result but should also demonstrate the method (each step) to calculate the checksum.

Problem Statement 4:

Write a C++/Java program that accepts an IP address and subnet mask in CIDR notation, and print the following information about the sub-network:

- 1) Subnet Mask in dotted decimal notation (Example 255.0.0.0)
 - 2) Network Address in dotted decimal notation: (Example 103.0.0.0)
 - 3) Usable Host IP Range: Starting IP 103.0.0.1 --- Ending IP 103.255.255.254
-

Problem Statement 6:

Write a C++/Java program for the Decibel (dB) calculator. Your program should perform the following operations:

- 1) If transmit power of a network device is given in watts (Example: '20 W') then the program should print the transmit power in Decibel Watts (dBW) and Decibel Milliwatts (dBm).
 - 2) Else if transmit power of a network device is given in Decibel Watts (dBW) or Decibel Milliwatts (Example: '20 dBW' or '20 dBm') then the program should print the transmit power in Watts.
-