This test is meant to help us evaluate your knowledge and design process. You have 1 hour 30 minutes to go through the questions but it is possible that you will not complete every step; this is normal.

During the test, you have full access to the Internet and you must ask questions at any time, if there is any confusion.

Your task is to write a simple C program to parse a pre-determined (provided at the top of c program file) UDP/IP/Ethernet packet and print each field separately. The parsing code needs to be able to parse the hard-coded packet buffer only, but it does not have to be capable of parsing all packet types. **However the program should not print incorrect data if a different packet type was given**. The program should verify the IP and UDP checksums and indicate if they are valid. Finally, the program should modify the packet with a **hardcoded IP address and Port (you can choose these values)**, as if the packet was being modified for NAT (Network Address Translation). The resulting packet should be properly formed.

To help you with this task, we have prepared a laptop running Windows and a Linux machine connected via ssh. Your program will need to be compiled using gcc under Linux.

ssh session is already connected to the Linux machine. In the "New_Hire" folder you will find a

skeleton for the main.c file, with the IP checksum and UDP checksum functions already available.

If you prefer to write your code using notepad, the drive Z: is mapped to the Linux m/c.

Please store all your work in the New_Hire folder of the Linux m/c.

To comiple your file :

gcc <filename> -o <executable-filename>

Example : gcc main.c -o main

Objective #1
Write pseudo code in 5 to 10 bullets or sub-bullets to describe the flow of the software. Save it in a file called pseudo.txt.

Objective #2
Open the main.c file from the Linux m/c, New_Hire folder. Update the file to parse the Ethernet packet and print the header content, at each layer (Ethernet MAC, IP, UDP). At this point you do not need to verify the checksums yet. Copy the result in main1.c.

Objective #3
Using the checksum functions provided in checksum.c, change the main.c file to verify that the IP and UDP checksums in the packet are correct. If the checksums turn out to be incorrect, do not try to fix this and simply move to the next step. Copy the result in main2.c.

Objective #4

In main.c, update the IP and UDP headers to replace the appropriate IP address and UDP port as if the packet was being modified for NAT. Once all necessary fields of the packet are updated, prove that the resulting packet remains valid. Copy the result in main3.c.

Good luck!