Ping and Watchdog

**הנחיות להגשת המטלה:**

את מטלה זו יש להגיש בזוגות כקובץ ZIP עם מספרי ת"ז של הסטודנטים/ות. לתיבת ההגשה במודל. הגשות באיחור יתאפשרו עד 4 ימים כאשר לכל יום איחור ירדו 5 נקודות. שימו לב, יש להגיש גם קובץ pdf המכיל צילומי מסך (בכל מקום שאתם עושים משהו תצלמו מסך ותסבירו איך הגעתם למסקנה) בכל מקום שאתם משדרים מידע צריך לצרף קובץ הקלטה. אין להגיש צילומי מסך של מרוכזים בקובץ אחד עם הסברים, קובץ ללא הסברים או צילומי מסך בלבד לא יבדקו

1. את המטלה יש להגיש עד התאריך המצוין בתיבת ההגשה
2. כל הקבצי המטלה (קוד, פלט תעבורה, הסבר) כולל הסברים שלכם והקלטות Wireshark דחוסים לקובץ ZIP ששמו הוא מס' ת.ז. של המגישים עם קו תחתון ביניהם ID\_ID.
3. הגשה בזוגות זהו הסטנדרט וזו חובה. לאישורים חריגים שלחו מייל למרצה שלכם ותכתבו את אחראי המתרגלים, אלמוג שור.
4. מותר לכם להשתמש בכל החומר שנמצא במודל כולל קוד בתרגולים. חומרים אחרים אין אפשרות. כמובן שאפשר להיעזר באינטרנט להבנה של תהליכים וקוד אבל בשום פנים ואופן לא להעתיק קוד. כל שימוש בכלים כמו copilot or chatgpt יפסלו
5. אין איחורים ללא אישור מיוחד של רכז הקורס (עמית), איחור ללא אישור יגרור אפס אוטומטי.
6. הגשת העבודות תתבצע דרך מערכת ה Moodle של הקורס (לא דרך האימייל(.
7. יש להקפיד על כללי עיצוב הקוד שנלמדו בתואר (נא להקפיד על פלט ברור, הערות קוד במידה ושמות משתנים בעלי משמעות). קוד רץ בלבד יכול לקבל לכל היותר ציון 60, שאר 40 הנקודות זה הסברים שלכם, ידע, קוד קריא וכו'.
8. ניתן להגיש תרגילים למערכת מספר בלתי מוגבל של פעמים כאשר כל הגשה דורסת את הקודמת.
9. העבודה הינה אישית של הזוג ואסור לקבל עזרה מאנשים מחוץ לאוניברסיטה או בתוכה לה. אנשים המתקשים ורוצים עזרה יכולים לפנות לצוות הקורס בשעות הקבלה או להעלות שאלה לאתר הקורס. עבדתם עם עוד זוג על בעיות מסוימות כתבו זאת בקובץ ה-pdf
10. אסור להעביר קטעי קוד בין סטודנטים, להעלות פתרונות או חלקי פתרונות לאתרים ברשת האינטרנט, פורומים או בקבוצות תקשורת שונות.
11. סטודנטים שיעתיקו פתרון, יקבלו 0 בכל המטלות בקורס ונעלה דיווח לוועדת המשמעת המוסדית.
12. קבצי הקוד המטלה נדרשים להתקמפל ולרוץ על מערכת הפעלה Ubuntu 22.04LTS
13. מסמך המתאר את המערכת והפתרון שלכם (המלצה, השתמשו במבנה המסמך ששלחנו לכם במודל)

## Myping - 60%

The ping command is used to check the connection between 2 machines. In this part, you will implement the "ping" command (see picture below).

****

You will write a program called "ping.py" which will get an argument indicating which host to ping.

Usage: python ping.py <ip> (just like the ping command)

The program will send an ICMP ECHO REQUEST to the host, and when receiving ICMP-ECHO-REPLY, the program will send the next ICMP ECHO REQUEST (no need to stop).

For each packet received, you will print the packet IP, packet sequence number, and time between the request and replay.

you can write your own code or modify the one in the Moodle.

\*In this part, handout ping.py code

## Watchdog timer 40%

Watchdog is a timer to detect and recover your computer dis-functions or hardware fails. It’s a chip whose sole purpose is to receive a signal every millisecond from the CPU. It will reboot the system if it hasn’t received any signal for 10 milliseconds (mostly when hardware fails).

Modify the ping program, and write a watchdog that will hold a timer (TCP connection on port 3000) to ensure that if we don’t receive an ICMP-ECHO-REPLY after sending an ICMP-REQUEST for 10 seconds, it will exit and print "server <ip> cannot be reached."

Modify the ping.py program so that it will execute the watchdog.py program as well using threads.

\*You don’t have to use threads. You can also use the fork(2) and exec(2) workflow for execution

\*Note that every time ping.py sends a packet, it will need to update watchdog.py timer.

\*It is required that the code will work on localhost on both ping.py and watchdog.py.

\*In part b you will handout better\_ping.py and watchdog.py