

# רשתות תקשורת – תשפ"ג – סמסטר א' – מטלה 4

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 קבצי קוד, pcaps וגם קובץ pdf המכיל צילומי מסך (בכל מקום שאתם עושים משהו תצלמו מסך ותסבירו איך הגעתם למסקנה כל פלט שיש לכם צריך להיות חלק מקובץ ההסבר).

- (1) את המטלה יש להגיש עד התאריך המצוין בתיבת הגשה.
- (2) יש להגיש את כל קבצי המטלה (קוד, פלט תעבורה, הסבר) כולל הסברים שלכם והקלטות Wireshark דחוסים **לקובץ zip** ששמו הוא מס' ת.ז. של המגישים עם **קו תחתון בניהם ID\_ID**.
- (3) 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.
- (4) אין איחורים ללא אישור מיוחד, איחור ללא אישור יגרור אפס אוטומטי.
- (5) הגשת העבודות תתבצע דרך מערכת ה Moodle של הקורס (לא דרך האימייל).
- (6) 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'.
- (7) ניתן להגיש תרגילים למערכת מספר בלתי מוגבל של פעמים כאשר כל הגשה דורסת את הקודמת.
- (8) 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.
- (9) אסור להעביר קטעי קוד בין סטודנטים, להעלות פתרונות או חלקי פתרונות לאתרים ברשת האינטרנט, פורומים או בקבוצות תקשורת שונות.
- (10) סטודנטים שיעתיקו פתרון, יקבלו 0 בכל המטלות בקורס ונעלה דיווח לוועדת המשמעת המוסדית.

## 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).

```
yuval@h-MacBook-Pro-s1-yyy ~ % ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8): 56 data bytes
64 bytes from 8.8.8.8: icmp_seq=0 ttl=115 time=11.696 ms
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=7.584 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=6.739 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=6.807 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=8.920 ms
```

You will write a program called “ping.c” which will get an argument indicating which host to ping.

usage: ./a.out <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 attached one\*\*\***

**\*in this part, handout ping.c code**

## Watchdog timer:

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.c program so that it will execute the watchdog.c program as well using fork + exec.

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

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

**\*in part b you will handout better ping.c and watchdog.c**