**עבודת הגשה מספר 2 – תקשורת ורשתות**

**מטרת העבודה**

הסטודנטים יתכננו, יממשו ויבחנו מערכת צ'אט מבוססת Sockets בשפת Python תוך שימוש בטכנולוגיות TCP, במטרה להבין את עקרונות התקשורת ברשת ובניית מערכות מבוזרות.

**ניתן להשתמש בכל שפה שתבחרו !**

**דרישות טכניות**

1. **תקשורת**
   * המערכת תשתמש בפרוטוקול TCP.
   * יש לממש תקשורת דו-כיוונית בין לקוחות (Clients) לשרת (Server).
   * השרת יטפל לפחות ב-5 לקוחות בו-זמנית.
2. **מבנה המערכת**
   * **שרת (Server):**
     + השרת יאזין לבקשות של לקוחות וישמש כמתווך בין הלקוחות.
     + כאשר השרת יקבל פניה לקוח השרת צריך לפתוח צ׳אט בין הלקוח ללקוח אחר. הפניה צריכה להכיל את שם הלקוח המבוקש. ניתן להניח שלכל לקוח יש שם יחודי.
   * **לקוחות (Clients):**
     + כל לקוח יוכל להתחבר לשרת, לשלוח ולקבל הודעות בזמן אמת.
     + הממשק יכול להיות טקסטואלי (אין צורך לממש ממשק גרפי).
3. **הגבלות טכניות**
   * אין להשתמש בספריות מוכנות לניהול שרתים/לקוחות כמו Spring, אלא לממש את כל התקשורת באמצעות Sockets בלבד.
   * אפשר להיעזר בדוגמא הזאת:

<https://pymotw.com/2/socket/tcp.html>

כדי לראות איך מתבצעת תקשורת שרת-לקוח בפרוטוקול tcp בפייתון.

* אפשר להיעזר בדוגמא הזאת:

<https://www.dunebook.com/creating-a-python-socket-server-with-multiple-clients/>

כדי לראות איך משתמשים בריבוי תהליכונים.

**דרישות קוד**

* הקוד צריך להיות קריא, מתועד ומחולק לקבצים ולמחלקות בצורה הגיונית.
* יש לכלול טיפול בשגיאות, לדוגמה:
  + ניתוק לא צפוי של לקוח.

**תוצרים להגשה**

1. **תיעוד**
   * הסבר כללי על המערכת ומבנה הקוד.
   * הוראות התקנה והרצה.
   * דוגמת קלט ופלט.
2. **קוד מקור**
   * יש להגיש את קבצי הקוד המלאים server client ואם יש עוד קבצים גם אותם
   * יש לצרף קובץ שמגדיר את מבנה בסיס הנתונים שנבחר.
   * יש לצרף הסבר מפורט איך להפעיל את השרת/לקוח

הסברים כללים

* שימו לב, כי המרצים בקורס יוכלו בכל עת לזמן את המפתחים לבדיקה פרונטלית, הימנעו מאי נעימות

בהצלחה!