



רשתות תקשורת מחשבים
סמסטר א' תשפ"ד

תרגיל בית מס' 1 להגשה עד 31.01.2024

חלק ראשון – TCP Client-Server Program (85%).

בתרגיל זה נכתוב תוכנת שרת-לקוח בסיסית בה לקוחות יוכלו להשתמש בשרת לצורך ביצוע מספר פעולות. השרת והלקוח ישוחחו בפרוטוקול אפליקציה אשר תגדירו בעצמכם, כאשר התעבורה תתבצע על גבי TCP. מטרת התרגיל:

- שימוש נכון בsocket programming, עבודה מול ממשקי רשת.
- תכנון ופיתוח פרוטוקול תקשורת ברמת האפליקציה.
- תמיכה בריבוי לקוחות במקביל.

השרת:

אתחול השרת מתבצע בשורת הפקודה (ולאחריה השרת ממתין ומשרת לקוחות).

```
./numbers_server.py users_file [port]
```

כאשר:

- users_file הוא path לקובץ טקסטואלי tab-delimited שמכיל שתי עמודות: שם משתמש וסיסמא (עבור אותו המשתמש). למשל:

Bob	simplepass
Alice	BetT3RpAas

כאשר בין שני השדות יש tab. ניתן להניח שאין מוסיפים לקוחות בעת ריצת השרת.

- הפורט להאזנה (port) הוא פרמטר אופציונלי עם ערך ברירת מחדל 1337.

לשם פשטות, מרגע שהשרת מתחיל לרוץ הוא אינו מסיים את ריצתו. השרת יוכל לשרת מספר לקוחות במקביל וירבב ביניהם בעזרת שימוש ב-select – כפי שנראה בתרגול. כלומר, בכל רגע ייתכנו מספר לקוחות המחוברים לשרת במקביל, והוא משרת את כולם. באותו הזמן, השרת מאזין ויכול לקבל לקוחות חדשים שמתחברים אליו. השימוש בselect יאפשר לשרת להימנע מקריאות חוסמות, ללא שימוש בmulti-threading.

הלקוח:

אתחול הלקוח מתבצע בשורת הפקודה:

```
./numbers_client.py [hostname] [port]
```

כאשר hostname וport הם פרמטרים אופציונליים שערכי ברירת המחדל שלהם הינם:

- hostname = localhost

- port = 1337

שימו לב, לא ניתן לספק port ללא hostname. hostname יכול להיות שם או כתובת IP.

התחברות:

כשלקוח חדש מתחבר לשרת, השרת שולח לו ברכת שלום:

Welcome! Please log in.

אחרי שהלקוח מקבל את ברכת השלום מהשרת, הוא מזין את פרטיו בפורמט הבא:



רשתות תקשורת מחשבים
סמסטר א' תשפ"ד

User: username_of_user

Password: password_of_user

הפרטים יישלחו לשרת. אם הם נכונים, השרת מקבל את הלקוח, מציג הודעה הכוללת את שם הלקוח:

Hi username_of_user, good to see you.

אחרת יודפס:

Failed to login.

ותינתן לו האפשרות להזין שוב את פרטיו.

פקודות נתמכות:

1. שליחת תרגיל חשבון פשוט, שתראה כך:

calculate: X Y Z

כאשר X ו-Z הם שני מספרים שלמים, ו-Y היא אחת מהפעולות הבאות: "+", "-", "x", "/". אשר מייצגות חיבור, חיסור, כפל וחילוק.

השרת יבצע את פעולת החישוב המתאימה ויחזיר את התשובה שתדופס כך:

response: R.

כשר R היא תוצאת החישוב, מספרית.

לשם פשטות – ניתן להניח כאן כי גודל הפרמטרים (X ו-Z) הוא עד גודל int.

2. בדיקה האם מספר הוא פלינדרום – האם המספר שהוכנס הוא פלינדרום. הפקודה תראה כך:

is_palindrome: X

השרת יענה: Yes או No, בהתאם לשאילתא שהוכנסה, ותודפס אחת מהאפשרויות:

response: Yes / No.

3. בדיקת ראשוניות של מספר – האם המספר ראשוני. הפקודה תראה כך:

is_primary: X

השרת יענה: Yes או No, בהתאם לשאילתא שהוכנסה, ותודפס אחת מהאפשרויות:

response: Yes / No.

דוגמת הרצה נמצאת בעמוד האחרון.

כמו כן, הלקוח יוכל להקליד quit על מנת לסגור את הסשן.

בכל שלב, במידה והלקוח לא עבר את שלב ההתאמתות ויישלח פקודה אחרת לשרת, השרת יסגור את החיבור מולו. כמו כן לשם פשטות ניתן להניח כי אם מתקבל קלט לא בפורמט צפוי מהמשתמש, התוכנה תדפיס הודעת שגיאה ותתנתק מהשרת.

דרישות התרגיל:

עליכם לממש את השרת והלקוח כך שיממשו את הפקודות והפעולות שהוגדרו בתרגיל. לשם כך, תתכננו פרוטוקול אפליקציה מתאים שיעבוד על גבי פרוטוקול TCP, וממשו אותו. את פרוטוקול האפליקציה יש לתעד בבירור, באופן שאמור לאפשר לכל אדם לממש לקוח או שרת ש"ידברו" עם התוכנות שהגשם. הגישו בנוסף קובץ Readme שבו תיאור של הפרוטוקול שבחרתם לממש.

הדגש בבדיקת הקוד שתגישו יינתן כמובן לעבודה מול socket (שימוש ב-API, בדיקת שגיאות, וכו') ועל מימוש התקשורת בין השרת והלקוח כך שיהיה יעיל והגיוני בהתאם לדגשים עליהם דיברנו בכיתה.



רשתות תקשורת מחשבים
סמסטר א' תשפ"ד

חלק שני – עבודה עם Wireshark (15%).

הורידו את תוכנת wireshark מהאתר <https://www.wireshark.org>. תוכנה זאת מאפשרת לצפות בפקטות שהתקבלו למחשב שלכם דרך אחד מכרטיסי הרשת הקיימים אצלכם, כפי שראינו בתרגול.

הריצו את התוכנה שכתבתם, בחרו את כרטיס הרשת הרלוונטי, השתמשו בfilter מתאים, והראו אלו פקטות עוברות בין הלקוח והשרת כאשר הלקוח מתחבר לשרת. צרפו קובץ המכיל תעבורה רלוונטית של הסשן שבו רואים את התעבורה הרלוונטית (יש ב-wireshark אפשרות של save as, בפורמט pcap).

כמו כן, בחרו חלק בתהליך התקשורת ביניהם לבחירתכם (לדוגמא - התחברות, או שליחת שאילתא מהלקוח, או קבלת תשובה מהשרת, וכו'). וצרפו print screen מתאים שמראה אותו ב-wireshark יחד עם תיאור קצר של התוכן/פרוטוקול אפליקציה שלכם שמועבר שם.

הנחיות הגשה:

הגשת התרגילים תתבצע דרך המודל.

הגישו קובץ ZIP יחיד ששמו:

EX1_ID1_ID2

אם הגשתם בזוג (רק אחד מבין השניים נדרש להגיש את התרגיל במודל), או

EX1_ID

אם הגשתם ביחיד.

קובץ ZIP יכול את כל קבצי הקוד, כך שיוכלו לרוץ בסביבת novan בהצלחה.

ניתן לחלק את הקוד לקבצים נוספים על קבצי ההרצה הראשיים, לפי שיקול דעתכם.

קבצי ההרצה יכולים להיות עם הסיומת py. או בלעדיה. רצוי להוסיף

```
#!/usr/bin/python3
```

בראשית הקבצים.

קובץ Readme.pdf שיכיל תיאור של הפרוטוקול ותכנון השרת והלקוח.

הקבצים הרלוונטיים לחלק השני (העבודה מול wireshark) ובהם קובץ pcap, ותמונות/PDF עם התעבורה הרלוונטית והסבר קצר.

 **בהצלחה**



רשתות תקשורת מחשבים
סמסטר א' תשפ"ד

דוגמת הרצה: (הטקסט **בכחול** – קלט מהמשתמש).

בצד השרת:

```
./numbers_server.py ~/my_dir/users_file.txt
```

בצד הלקוח:

```
./numbers_client.py
```

Welcome! Please log in.

User: Bob

Password: simplepass

Hi Bob, good to see you.

is_primary: 382807

response: Yes.

כעת יתחבר לקוח נוסף:

```
./numbers_client.py
```

Welcome! Please log in.

User: Alice

Password: BetT3RpAas

Hi Alice, good to see you.

is_palindrome: 12345678987654322

response: No.

quit

בוב ימשיך בפעילות במקביל:

calculate: 34 + 837

response: 871.

calculate: 212121 / 837

response: 253.43.

quit