

## תרגיל מעשי 1

תיעוד פרוטוקול Barak-Neufeld

פלג נויפלד 308401108

יובל ברק 203769609 yuvalyehudab@mail.tau.ac.il

חשוב(!): בסוף הקובץ יש הסבר על מבנה הקבצים. בנינו אותם כדי להפחית שכפול קוד ולאפשר בדיקה נוחה של כל קובץ שנפרד, ויצאו הרבה קבצים. יש הסבר ואיור למטה.

א. מבנה פקטות מהלקוח לשרת:

- פקטה של פקודת Login:
  - שני בייטים ובהם אורך שם המשתמש בבייטים
  - לאחר מכן מופיעים הבייטים ובהם שם המשתמש
  - שני בייטים ובהם אורך הסיסמה בבייטים
  - לאחר מכן מופיעים הבייטים ובהם הסיסמה
  - הלקוח מקבל מהשרת פקטה בת בייט בודד: '1' עבור התחברות מוצלחת, ו-'0' עבור כישלון בהתחברות.
- פקטות של פקודות ניהול תוכן:
  - כל פקודה שולחת מספר פקטות משתנה. בפקטה הראשונה שנשלחת מופיע קוד הפקודה, ובפקטות הבאות מופיעים הפרמטרים של הפקודה, באופן המוסבר להלן:

0. פקודת QUIT

- פקטה של קוד פקודה - בת בייט אחד ובו '0'

1. פקודת LIST OF COURSES

- פקטה של קוד פקודה - בת בייט אחד ובו '1'

2. פקודת ADD COURSE

- פקטה של קוד פקודה - בת בייט אחד ובו '2', ולאחר מכן:
- פקטה המורכבת מ-2 בייטים ובהם אורך מספר הקורס ( $N$ ) ולאחר מכן  $N$  בייטים עם מספר הקורס עצמו, לאחר מכן 2 בייטים ובהם אורך שם הקורס ( $M$ ) ולאחר מכן  $M$  בייטים עם שם הקורס עצמו.
- הלקוח מקבל מהשרת פקטה בת בייט בודד: '1' אם הקורס כבר קיים במאגר, ו-'0' אחרת.

3. פקודת RATE COURSE

- פקטה של קוד פקודה - בת בייט אחד ובו '3', ולאחר מכן:
- פקטה המורכבת מ-2 בייטים ובהם אורך מספר הקורס ( $N$ ) ולאחר מכן  $N$  בייטים עם מספר הקורס עצמו, לאחר מכן 2 בייטים ובהם אורך הציון המספרי ( $M$ ) ולאחר מכן  $M$  בייטים עם הציון המספרי עצמו. לאחר מכן 2 בייטים ובהם אורך הדירוג הטקסטואלי ( $L$ ) ולאחר מכן  $L$  בייטים עם הדירוג הטקסטואלי עצמו.

#### 4. פקודת GET RATE

- פקטה של קוד פקודה - בת בייט אחד ובו '4'
- פקטה המורכבת מ-2 בייטים ובהם אורך מספר הקורס (N=) ולאחר מכן N בייטים עם מספר הקורס עצמו.

ב. מבנה של פקודות מהשרת ללקוח:

- פקטה של פקודת Login - בת בייט בודד, ובו 0 עבור כישלון (אין לקוח עם שם משתמש וסיסמה כפי שנשלחו), 1 עבור הצלחה.
- פקטות פקודות ניהול תוכן
- פקטה של list of courses ::

#### 0. פקודת QUIT

- השרת מקבל את הפקטה ומבצע סגירה של החיבור ללקוח (אין שליחה של פקטה ללקוח).

#### 1. פקודת LIST OF COURSES

- פקטה ובה שני בייטים של אורך השורה, ואז תוכן השורה.
- אם אורך השורה הוא 0 - סיימנו.

#### 2. פקודת ADD COURSE

- פקטה בת בייט אחד ובו '1' אם הקורס המבוקש כבר קיים במאגר, ו-'0' אחרת.

#### 3. פקודת RATE COURSE

- אין שליחת פקטות ללקוח בפקודה זו.

#### 4. פקודת GET RATE

- פקטה ובה שני בייטים של אורך השורה, ואז תוכן השורה. בדומה לפקודת LIST OF COURSES.

הפרוטוקול מניח את ההנחות הבאות -

- א. הפורט הדיפולטי הוא 1337 (עבור השרת והלקוח)
- ב. אורך ה-hostname - לכל היותר 1024 תווים
- ג. מספר הלקוחות השונים שיפנו לשרת במשך פעולתו הוא לכל היותר 32. אלה יכולים להתחבר לשרת מספר לא מוגבל של פעמים בכל סדר שהוא.
- ד. שם של יוזר הוא לכל היותר 16 תווים. אורך סיסמה - לכל היותר 16 תווים.
- ה. תיאור מקסימלי של קורס - 2048 תווים
- ו. אורך מקסימלי של מספר קורס - 8 תווים
- ז. אורך מקסימלי של דירוג מספרי של קורס - 4 תווים
- ח. אורך פקודה מקסימלי של לקוח - 4096 תווים
- ט. אורך פקטה שנשלחת לשרת מהלקוח - לכל היותר 8192 תווים
- י. אורך תור הבקשות המקסימלי של השרת - 16
- יא. אורך ניתוב קובץ מקסימלי - 1024 תווים

## הנחות נוספות:

1. קבצים תקינים - בפורמט הנכון
2. המשתמש מעביר ארגומנטים לא ריקים - שם משתמש + סיסמא

## מבנה התוכנית: (איור – בסוף)

### לקוח:

1. קובץ ראשי – `seker_client`. 3 פונקציות עיקריות: התחברות לשרת, שליחה ואישור של שם משתמש וסיסמא, הרצת פקודות במקרה של הצלחה בחיבור (שם משתמש וסיסמא נכונים). משתמש ב-2 הקבצים הבאים:
2. קובץ התחברות – `client_connection`. יצירת סוקט כפי שנלמד בכיתה.
3. קובץ זיהוי פקודה ופרמטרים ושליחה לטיפול מתאים – `client_manager`. 2 פונ' עיקריות: הכנת אימות מול השרת וזיהוי פקודות ופרמטרים במהלך פעילות חיבור שאומת. משתמש בקובץ הבא:
4. קובץ `client_parser`. מימוש הפרוטוקול – שליחה וקבלה של מחרוזות לפי הפרוטוקול למעלה. 6 פונ' עיקריות – אימות פרטים והפונ' שנדרשנו לממש בתרגיל (הוספה, דירוג, יציאה, צפייה ברשימת הקורסים וצפייה בדירוג קורס). משתמש בקובץ הבא:
5. קובץ לשליחה של חוצץ – `full_connection`. מתמודד עם שליחה חלקית, משמש גם בצד השרת וגם בצד הלקוח – חבל לכתוב פעמיים. 2 פונ' – שליחה וקבלה של חוצץ.

### שרת:

1. קובץ ראשי – `seker_server`. טוען את המשתמשים למערך, יוצר סוקט האזנה, ואז באופן סדרתי מקבל לקוח, מאמת פרטים (לא מאפשר גישה עד שליחה מאומתת) ואז מקבל פקודה ושולח תשובה. משתמש ב – 3 קבצים:
2. קובץ טעינת משתמשים מאחסון – `load_users_from_file`. פונ' אחת שמקבלת מערך וקובץ וממלאת את המערך מהקובץ לפי הפורמט מהתרגיל.
3. קובץ התחברות – `server_create_sockets`. 2 פונ' עיקריות, ליצירת סוקטים כפי שנלמד בכיתה (עבור שרת - צריך 2 סוקטים).
4. קובץ עבודה מול הלקוח – `server_manager`. 2 פונ' עיקריות – אימות פרטי לקוח, וסיווג ושליחה של בקשות לטיפול. משתמש ב – 3 קבצים:
5. קובץ שליחה של חוצץ – מספר 5 למעלה, אין צורך להסביר שוב.
6. קובץ פונ' עזר – `server_utils`. 3 פונ' קטנות: מציאה של לקוח ע"פ שם וסיסמא במערך שנטען בתחילת התכנית, חילוץ פרמטר ללא גרשיים (לדוג' מספר קורס, שם משתמש) וחילוץ פרמטר עם גרשיים (לדוג' שם קורס/תיאור דירוג). גם קובץ זה משתמש קובץ השליחה (`full_connection`) הידוע בבינוי 5.
7. קובץ מימוש הפרוטוקול וביצוע הפעולות – `server_parser`. 4 פונ' עיקריות: הוספת קורס, דירוג קורס, שליחת רשימת הקורסים ושליחת דירוג של קובץ ספציפי. מנהל את קבלת המידע, השמירה בקבצים לפי הפורמט בחירת התשובה ללקוח והכנת החוצצים לשליחה. משתמש גם ב – 2 הקבצים שלפניו.



