

word server - 13

13.1 סקירה כללית

בפרק הזה תעבוד עם שרת ששולח זרם של נתונים מקודדים, ספציפית חבילות (packets) המכילות מילים אקראיות. תצטרך לכתוב קוד שיקבע מתי קיבלת חבילה שלמה וידפיס את הנתונים.

- **קבצים להורדה:**
 - `wordserver.py`: שרת ששולח מילים אקראיות.
 - `wordclient.py`: קוד שלד עבור הלקוח.
- **מגבלות:** אל תשנה קוד קיים, אלא אם כן יש סימן `TODO`. אתה יכול להוסיף פונקציות נוספות אם תצטרך.
- **דרישות:**
 - הקוד צריך לפעול עם כל ערך שמועבר ל-`recv()` בין 1 ל-4096.
 - יש להתמודד עם אורך מילים מ-1 עד 65535. השרת לא ישלח מילים מאוד ארוכות, אך תוכל לבדוק עם אורך מילים שונה.

13.2 מהי "חבילה של מילה"?

חבילה של מילה מורכבת מאורך המילה (מקודד כמספר 2-בייט ב-big-endian) ואחריו המילה המקודדת ב-UTF-8. מבנה החבילה הוא ככה:

- **אורך:** 2 בייטים, מקודדים כמספר ב-big-endian שמציין את האורך של המילה בבתים.
- **מילה:** תו מקודד ב-UTF-8 שמייצג את המילה.

לדוגמה:

- **מילה:** "hello"
 - **אורך:** 5 (מקודד כ-05 00)
 - **קידוד UTF-8:** 68 65 6C 6C 6F

חבילה למילים "hi" ו-"encyclopedia" תיראה כך:

- **חבילה:** 00 02 68 69 00 61 69 64 65 70 6F 6C 63 79 6E 65 0C
 - המילה הראשונה: "hi" (אורך 2)
 - המילה השנייה: "encyclopedia" (אורך 12)

13.3 יישום: `get_next_word_packet()`

הפונקציה הזו מקבלת את החיבור (socket) כארגומנט ומביאה את החבילה הבאה של מילה שלמה. היא פועלת על ידי קריאת זרם הנתונים, חילוץ אורך המילה, ואז קבלת המילה המתאימה.

- **תהליך:**
 1. קרא את 2 הבייטים הראשונים כדי לקבוע את אורך המילה.
 2. השתמש ב-`recv()` כדי לקבל את מספר הבייטים הזה מהזרם.
 3. החזר את החבילה המלאה (האורך + המילה).

4. אם דרושים נתונים נוספים, חזור על התהליך.

לדוגמה, אם התקבלו 5 הבייטים הראשונים (00 02 68 69 00), תוכל לחלץ את המילה הראשונה "hi" ולהחזיר את החבילה המלאה.

13.4 יישום: `extract_word()`

הפונקציה `extract_word()` אחראית לחילוץ המילה מתוך החבילה המלאה. החבילה מורכבת משני חלקים:

1. **בייטים של אורך** (2 בייטים, ב-big-endian).
2. **המילה המקודדת ב-UTF-8**.

כדי לחלץ את המילה:

1. חתוך את החבילה כדי לקבל את נתוני המילה (פסול את הבייטים של האורך).
2. השתמש בשיטה `decode()` כדי להמיר את נתוני הבייטים בחזרה למיתר (string) ב-UTF-8.

לדוגמה:

- **חבילה קלט:** 00 02 68 69
- **פלט:** "hi"

הערות חשובות

- **התמודדות עם סדר בייטים:** אורך המילה מקודד כמספר 2-בייט ב-big-endian. ודא שאתה מפרש נכון את סדר הבייטים.
- **ניהול זיכרון:** ייתכן שתצטרך לאחסן את הזרם ולעבד אותו בהדרגה כדי להתמודד עם חבילות לא שלמות.
- **בדיקות:** בצע בדיקות עם ערכים שונים שמועברים ל-`recv()` ואורכים שונים של מילים כדי לוודא שהקוד שלך מתמודד עם מגוון קלטים.

הפרק הזה נועד ללמד אותך כיצד להתמודד עם תקשורת מבוססת חבילות, במיוחד כיצד להתמודד עם זרמים של נתונים ולחלץ מהם מידע משמעותי.