

תרגיל בית מעשי מס' 1

להגשה עד 11.11.2013

שימו לב: באתר הקורס תוכלו למצוא מסמך עם הנחיות מפורטות לגבי הגשת תרגילים מעשיים. על כל התרגילים המעשיים שתגישו הסמסטר לעמוד בהנחיות הללו.

מטרה

בתרגיל זה נממש שרת פשוט להעברת קבצים ונכתוב לקוח עבורו. השרת והלקוח ישוחחו בפרוטוקול אפליקציה אשר תגדירו בעצמכם.

בתרגיל מעשי מספר 3 נרחיב את האפליקציה ונהפוך אותה מ – server-based למבוזרת.

השרת

השרת מחזיק אוסף של קבצים. הוא מאפשר ללקוח אחד בכל פעם להתחבר אליו, למשוך קבצים או לשלוח קבצים לשרת.

אתחול השרת מתבצע בשורת הפקודה

```
file_server dir [port]
```

כש - dir היא ספרייה לוקאלית שבה השרת מחזיק את הקבצים שהלקוחות יכולים למשוך, ואליה הוא ישמור קבצים שהלקוחות ישלחו אליו. לשם פשטות, אתם יכולים להניח שהספרייה הזו היא נתיב אבסולוטי שאינו מכיל את התו ~ ושהיא נגמרת בתו /. למשל, על מחשבי המעבדה:

```
/specific/a/home/cc/students/cs/my_username/my_dir/
```

אפשר להניח שהספרייה קיימת ואין צורך ליצור אותה.

הפורט להקשבה הוא פרמטר אופציונלי עם ערך ברירת מחדל 2533.

לאחר מכן השרת ממתין ומשרת לקוחות. לשם פשטות, מרגע שהשרת מתחיל לרוץ הוא אינו מסיים את ריצתו. בכל רגע נתון הוא נדרש לטפל רק בלקוח אחד (ולכן מותר לכם להשתמש בפעולות חוסמות). מצד שני, הוא צריך להיות מוכן לשרת מספר רב של לקוחות לאורך חייו (כל עוד הם לא מנסים להתחבר באותו הזמן).

כשלקוח חדש מתחבר לשרת השרת שולח לו ברכה כלשהי, שהיא שורת טקסט לבחירתכם. למשל:

```
Welcome! I am simple-file-server.
```

לאחר מכן הוא ממתין ללקוח שיתחבר ויכניס פקודות.

הלקוח

אתחול הלקוח מתבצע בשורת הפקודה:

רשתות תקשורת מחשבים, סמסטר א' 2013/14
ביה"ס למדעי המחשב, אוניברסיטת ת"א

```
file_client dir [hostname [port]]
```

כאשר hostname ו-port הם פרמטרים אופציונאליים. ערך ברירת המחדל הוא hostname = localhost, port = 2533. לא ניתן לספק port ללא hostname. הפרמטר hostname יכול להיות שם או כתובת IP.

dir היא הספרייה שבה הלקוח שומר את הקבצים שהוא מקבל מהשרת, וקבצים ממנה הוא יכול לשלוח לשרת. אפשר להניח לגבי הפרמטר הזה את אותן הנחות שאפשר להניח לגבי הפרמטר dir של השרת.

הלקוח יתחבר לשרת לפי הפרמטרים hostname, port. אחרי שהלקוח מקבל את הודעת הברכה של השרת הוא קולט פקודות מהמשתמש.

המשתמש יכול להכניס את הפקודות הבאות:

1. DIR

בתגובה, השרת ישלח למשתמש רשימה של הקבצים שמאוחסנים אצלו. תוכנת הלקוח תדפיס אותם, כל קובץ בשורה חדשה.

2. GET filename

השרת ישלח ללקוח את הקובץ ששמו filename. כשהפעולה תושלם בהצלחה יודפס ללקוח

```
transfer complete
```

3. PUT filename

הלקוח ישלח לשרת את הקובץ ששמו filename מתוך הספרייה של הלקוח, והשרת יאחסן את הקובץ הזה בספרייה של השרת. כשהפעולה תושלם בהצלחה יודפס ללקוח

```
transfer complete
```

גם בקבלה וגם בשליחה, אם קובץ באותו שם כבר קיים בספריית היעד, הקובץ החדש ידרוס את הקובץ הישן.

4. QUIT

יציאה מהתוכנה

לשם פשטות אפשר להניח:

- שמות הקבצים אינם מכילים whitespaces (רווח, \t וכו').
- שמות הקבצים הם case-sensitive.

דוגמת ריצה

בצד השרת:

```
file_server /specific/a/home/cc/students/cs/my_username/dir1/  
בצד הלקוח (בפונט ירוק – קלט מהמשתמש):
```

```
file_client /specific/a/home/cc/students/cs/my_username/dir2/  
Welcome! I am simple-file-server.
```

```
DIR
```

```
hello_world.txt
```

```
MyExcelSheet.xlsx
```

```
MyFavoriteSong.mp3
```

```
GET MyFavoriteSong.mp3
```

```
transfer complete
```

```
PUT AnotherGreatSong.mp3
```

```
transfer complete
```

```
QUIT
```

(שימו לב שהנחנו שבספרייה של הלקוח יש קובץ בשם AnotherGreatSong.mp3).

כעת מתחבר לקוח שני:

```
file_client /specific/a/home/cc/students/cs/other_user/dir/  
Welcome! I am simple-mail-server.
```

```
DIR
```

```
hello_world.txt
```

```
MyExcelSheet.xlsx
```

```
MyFavoriteSong.mp3
```

```
AnotherGreatSong.mp3
```

```
GET AnotherGreatSong.mp3
```

```
transfer complete
```

```
QUIT
```

(שימו לב שהלקוח הראשון כבר שלח את הקובץ AnotherGreatSong.mp3 לספריית השרת, ולכן השרת מציג את הקובץ והוא זמין למשיכה ללקוחות שמתחברים אחר כך)

דרישות התרגיל

ראשית, עליכם לתכנן פרוטוקול אפליקציה מתאים שיעבוד מעל TCP. לאחר מכן, ממשו אותו כפי שנלמד בתרגול. האפליקציה סינכרונית, כלומר אפשר להשתמש בפקודות חוסמות.

את פרוטוקול האפליקציה יש לתעד בבירור, באופן שיאפשר לכל אדם לממש לקוח או שרת ש"ידברו" עם התוכנות שהגשתם.

הדגש בבדיקת הקוד שתגישו ינתן כמובן למימוש התקשורת ברשת. על המימוש להיות יעיל ורובוסטי (robust). אל תשכחו לבדוק שגיאות בערכי חזרה מפונקציות ולטפל בהם בהתאם.

למקרה שתבחרו להגיש בסביבת nova: ייתכן שזוגות רבים יבדקו את הפתרון שלהם על nova בו-זמנית. לכן, מומלץ להשתמש בפורט שרת שאינו פורט ברירת המחדל בעת בדיקת הקוד.