TEL AVIV UNIVERSITY אוניברסיטת תל-אביב

רשתות תקשורת מחשבים
סמסטר א' 2014/15

תרגיל בית מעשי מס' 1

להגשה עד 23.11.2014

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

מטרה

בתרגיל זה נתכנן פרוטוקול אפליקציה ונממש אפליקציית רשת במבנה שרת/לקוח. האפליקציה תממש גרסה פשוטה של המשחק Nim:

http://en.wikipedia.org/wiki/Nim

פתרון התרגיל מורכב משתי תוכנות. הלקוח (client) – מאפשר משחק אינטראקטיבי לשחקן בודד .

השרת (server) – מנהל את המשחק, ומשחק מול השחקן שמשתמש בתוכנת הלקוח.

בתרגיל מעשי מספר 2 נרחיב את האפליקציה כך שתתמוך במספר שחקנים.

המשחק

בסעיף זה נתאר את חוקי המשחק עבור גרסה פשוטה של Nim.

המשחק מיועד לשני משתתפים. למשחק יש פרמטר M (תוכלו להניח $1 \le M \le 1500$) ופרמטר בוליאני IsMisere.

בתחילת המשחק, ישנן 4 ערימות (heaps) A,B,C,D המכילות M קוביות.

בכל תור לוקח אחד המשתתפים מס' קוביות (לפחות קובייה אחת) כרצונו מאחת הערימות.

במשחק רגיל (isMisere == false) המנצח הוא האחרון שלוקח קוביות מהלוח. במשחק Misere (כלומר isMisere == true) המפסיד הוא האחרון שלוקח קוביות מהלוח.

להלן דוגמה למשחק רגיל, בו isMisere == false ו-$M = 5$.

A	B	C	D	
5	5	5	5	Alice takes 4 from A

5	5	5	1	Bob takes 1 from B
5	4	5	1	Alice takes 2 from A
3	4	5	1	Bob takes 2 from A
1	4	5	1	Alice takes 3 from C
1	4	2	1	Bob takes 1 from B
1	3	2	1	Alice takes 1 from B
1	2	2	1	Bob takes entire A heap.
0	2	2	1	Alice takes 1 from B
0	1	2	1	Bob takes 1 from C
0	1	1	1	Alice takes entire B heap.
0	0	1	1	Bob takes entire C heap.
0	0	0	1	Alice takes entire D heap and wins.

הלקוח

תוכנית הלקוח מיועד לתקשורת בין השחקן לבין תוכנית השרת. הלוגיקה שבה מיועדת אך ורק לתקשורת עם המשתמש ועם תוכנית השרת. למשל, התוכנית לא "זוכרת" את גדלי הערימות של המשחק.

שורת הפקודה להרצה היא:

nim [hostname [port]]

כאשר hostname ו-port הם פרמטרים אופציונאליים. ערך ברירת המחדל הוא hostname = localhost, ו-port = 6325. לא ניתן לספק port ללא hostname. הפרמטר hostname יכול להיות שם או כתובת IP.

למשל:

nim nova.cs.tau.ac.il 45678

הלקוח מתחבר לשרת בכתובת והפורט הנתונים.

עם ההתחברות, הלקוח מקבל מהשרת את הערך של IsMisere ומציג אותו למשתמש באופן הבא:

This is a Misere game

או לחילופין:

This is a Regular game

במהלך המשחק הלקוח מבצע את הצעדים הבאים אחד אחרי השני בלולאה, עד לסיום המשחק:

1. קבלת עדכון מהשרת לגבי גדלי הערימות הנוכחיים. יודפס:

Heap sizes are #, #, #, #

(כאשר # מוחלף בערכים המתאימים).

2. אם המשחק נגמר, הלקוח מקבל חיווי מהשרת לגבי זהות המנצח. יודפס:

I win! –OR- You win!

אחרת, הלקוח מתבקש להזין את הצעד הבא שלו לconsole. במקרה זה יודפס:

Your turn:

הקלט הוא תו המייצג את הערימה הנבחרת (A, B, C, D) או Q עבור Quit ליציאה מהמשחק. אם הקלט אינו Quit התוכנית מקבלת גם מספר שלם המייצג את מספר הקוביות שברצון הלקוח להוריד מהערימה שבחר.

3. אם הקלט היה Q, תוכנת הלקוח מסיימת את ריצתה מייד, אחרת הצעד של השחקן נשלח לשרת.

4. הלקוח מקבל אישור מהשרת שהמהלך נקלט, או לחילופין התראה שהמהלך היה לא חוקי :

למשל הלקוח ניסה להוריד 5 קוביות מערימה שהיו בה רק 3 קוביות.

במקרה של מהלך לא חוקי, הלקוח מפסיד את התור, ולא מקבל הזדמנות להכניס קלט נוסף. יודפס:

Illegal move –OR- Move accepted

5. בכל שלב על הלקוח לזהות גם ניתוק מהשרת. במקרה כזה, המשחק מסתיים ומודפס:

Disconnected from server

ההדפסות למסך חייבות להיות בדיוק כפי שמתואר בסעיף זה.

השרת

השרת מנהל משחק בודד. השרת "זוכר" את גדלי הערימות, מקבל צעדי משחק מהלקוח, ומשחק אחרי

כל צעד של הלקוח.

שורת הפקודה להרצה היא:

nim-server M isMisere [port]

כאשר M וisMisere הם הפרמטרים למשחק המתוארים לעיל. isMisere יקבל 1 או 0. הפורט

להקשבה הוא פרמטר אופציונלי עם ערך ברירת מחדל 6325 .

למשל:

nim-server 12 0 45678

לאחר שאותחל, השרת מציב את גדלי הערימות התחיליים ומחכה להתחברות מהלקוח .

לאחר התחברות המשחק מתחיל. השרת מגיב להודעות מהלקוח כפי שפורט בסעיף הקודם.

אחרי תורו של הלקוח, אם המשחק לא הסתיים מגיע תורו של השרת.

לוגיקה של השרת

הלוגיקה של השרת תהיה פשוטה למימוש:

בכל שלב השרת יוריד קובייה אחת מהערימה עם הכי הרבה קוביות.

אם יש כמה ערימות עם אותו מספר של קוביות- נוריד קובייה מהערימה עם האינדקס הגבוהה ביותר.

תוכנית השרת מסיימת את ריצתה לאחר שהודיעה ללקוח על סוף המשחק, או לאחר שהלקוח התנתק.

דוגמת ריצה

בצד השרת:

nim-server 2 0

בצד הלקוח (בפונט ירוק – קלט מהמשתמש):

nim

This is a Regular game

Heap sizes are 2, 2, 2, 2

Your turn:

A 2

Move accepted

Heap sizes are 0, 2, 2, 1

Your turn:

A 5

Illegal move

Heap sizes are 0, 2, 1, 1

Your turn:

B 2

Move accepted

Heap sizes are 0, 0, 1, 0

Your turn:

C 1

Move accepted

Heap sizes are 0, 0, 0, 0

You win!

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.

TEL AVIV UNIVERSITY אוניברסיטת תל-אביב

רשתות תקשורת מחשבים

סמסטר א' 2014/15

בהצלחה :)