

מבוא לתכנות מערכות תרגיל בית מספר 0

סמסטר חורף 2014/15

תאריך פרסום: 23/03/2015

תאריך הגשה: 31/03/2015

מתרגל אחראי לתרגיל: דני רסין 234122cs@gmail.com

1 הערות כלליות

- תרגיל זה מהווה 1% מהציון הסופי
- בכל השאלות בנוגע לתרגיל, יש לפנות לדני (ולא למתרגל אחר). בכל פנייה בקשר לתרגיל, נא לכתוב בשורת הנושא (Subject): mtm0.

2 הקדמה

מטרת תרגיל זה היא ביצוע מספר צעדים ראשונים בעבודה מרחוק על שרתים ובסביבת UNIX. התרגיל מורכב משלושה חלקים:

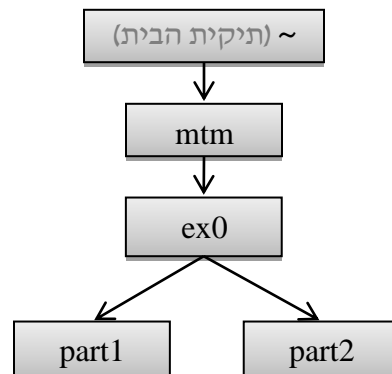
1. התחברות ל-t2 וביצוע פעולות בסיסיות.
2. כתיבת תכנית ראשונה ב-C, הידורה ובדיקתה על שרת ה-t2.
3. מציאת באגים בתכנית לדוגמה ע"י שימוש בדיבאגר.

הערות:

- באתר הקורס מופיעים מדריכים המסבירים כיצד להתחבר ל-t2.
- התרגיל להגשה **ביחידים**.
- יש להגיש את חלק ב' וג' של התרגיל כך שייבדק על ידי הבודק האוטומטי אשר בשימוש בקורס.

3 חלק א' - התחברות ופעולות בסיסיות

1. התחברו לחשבונכם בשרת ה-t2. הוראות מפורטות על הדרכים השונות לעשות זאת נמצאות במדריך ההתחברות ל-t2 שבאתר תחת Course Material או בקישור הבא: http://webcourse.cs.technion.ac.il/234122/Spring2015/ho/WCFiles/Connecting_to_t2.pdf
2. לאחר ההתחברות צרו את התיקיות הבאות תחת תיקית הבית שלכם: תיקיה בשם mtm, בתוכה תיקיה בשם ex0 ובתוכה תיקיות בשמות part1 ו-part2. כך שיווצר המבנה הבא:



3. כעת נרצה להעתיק את הקבצים הדרושים לפתרון שאר חלקי התרגיל מחשבון הקורס לחשבונכם האישי. לשם כך העתיקו את כל הקבצים הנמצאים תחת התיקיה ~mtm/public/1415b/ex0/part1

4 חלק ב' - תכנית ראשונה ב-t2

4.1 מפרט התכנית

4.2 דגשים והמלצות

- ### 4.3 הידור ובדיקה

אם הקבצים זהים לא יודפס כלום, אם יש הבדל יודפסו ההבדלים בין הקבצים.

בנוסף, מסופק לכם קובץ בשם `mtm_sol`, המהווה גרסה מקומפלת של התכנית אותה אתם צריכים לכתוב. ניתן להשתמש בו כדי לבדוק מקרים נוספים ולייצר טסטים נוספים.

5 חלק ג' - דיבוג

נעבור כעת לתיקיה `part2` שיצרנו בחלק א'. תיקיה זו מכילה קובץ קוד בשם `mtm_buggy.c` וקבצי בדיקה ופלט צפויים.

1. התכנית `mtm_buggy` אמורה לקלוט מהמשתמש מספר מחרוזות (בדומה לתכנית בחלק הקודם) ולאחר מכן להדפיס את המחרוזות הארוכה ביותר, המחרוזות המינימלית לפי סדר לקסיקוגרפי ואת המחרוזות המקסימלית לפי סדר זה.
2. הדרו את התכנית (לא לשכוח את כל הדגלים) ונסו להריץ את התכנית עם קובץ הבדיקה הראשון. התכנית מתרסקת בשגיאת "Segmentation fault", משמעות השגיאה היא שהתכנית מנסה לקרוא ערכים מתאי זיכרון שאינם מוקצים לה. בד"כ שגיאות אלה נובעות ישירות משימוש לא נכון במצביעים או פשוט נסיון לקרוא מצביע שערכו NULL.
- אמנם הקוד בתכנית `mtm_buggy.c` אינו גדול במיוחד, אך כבר בכמות כזו של קוד יש להשקיע זמן מה למציאת הנקודה בה מתרחשת השגיאה. כדי למצוא את השגיאה הזו בקלות ניתן להשתמש בדיבאגר `gdb` אשר מותקן על שרת ה-`t2`.
- הריצו את הפקודה הבאה אשר מתחילה את הדיבאגר עם התכנית `mtm_buggy` (כאשר `mtm_buggy` הוא שם קובץ ההרצה)

```
> gdb mtm_buggy
```

- `gdb` הוא דיבאגר העובד בטרמינל. כדי להשתמש בו יש להכניס פקודות בדומה לשימוש הרגיל בטרמינל. כדי להריץ את התכנית ניתן להשתמש בפקודה `run` (כדי להפנות קלט ופלט פשוט מוסיפים את ההפניות כמו בד"כ). נסו להריץ את התכנית עם קובץ הקלט מתוך `gdb`.
- התכנית רצה תחת הדיבאגר כמו בריצה רגילה, אך הפעם כאשר נגיע לגישה הלא חוקית הדיבאגר יעצור את התכנית ויודיע על השגיאה. בשלב זה נוכל להשתמש למשל בפקודה `bt` כדי להדפיס את מצב מחסנית הקריאות. פקודות נוספות ניתן ללמוד פשוט ע"י שימוש בפקודה `help`.
- עדיין קיימת בעיה המקשה עלינו: המידע במחסנית הקריאות אינו מפורט מספיק. כדי לאפשר ל-`gdb` להדפיס מידע מדויק יותר יש להדר מחדש את התכנית ולהוסיף את הדגל `-g`. דגל זה שומר מידע עבור דיבאגרים בתכנית ומאפשר להם להתייחס לקוד המקור.
- צאו מהדיבאגר (ע"י הפקודה `quit`), הדרו מחדש את התכנית והריצו אותה תחת `gdb`. הפעם כאשר תדפיסו את מצב המחסנית תקבלו פירוט של השורות בקוד מהן התבצעו הקריאות לכל פונקציה. למעשה, תקבלו את השורה המדויקת בה קרתה השגיאה. שימו לב ששורה זו היא חלק מפונקציה שמימושה לא נתון ולכן עליכם לחפש את הבאג בפונקציות הקוראות לה.
- תקנו את השגיאה.
- לאחר תיקון השגיאה הריצו שוב את הקוד ותיווכחו לדעת שקיימת עוד שגיאה בקוד. מצאו ותקנו גם אותה.
- כעת התכנית עובדת נכון עם הדוגמה הראשונה - אך זה אינו מבטיח את נכונותה. הריצו את התכנית עם קובץ הבדיקה השני ומצאו את השגיאה הנוספת שהוא חושף - לולאה אינסופית. (הערה: כדי לעצור תכנית שנתקעה בלולאה אינסופית ניתן לחוץ על `Ctrl+C`).
- לאחר תיקון כל שלושת השגיאות ודאו שהתכנית מוציאה פלט זהה לזה שבקבצי הפלט.

6 הגשה

יש להגיש את חלק ב' וג' בהגשה אלקטרונית. תוכלו לנצל הזדמנות זו להיכרות עם הבודק האוטומטי ולחסוך אי-נעימויות בתרגילים הבאים.

את ההגשה האלקטרונית יש לבצע דרך אתר הקורס. תחת `Assignments`, `Exercise 0`, `Electronic submission`. קובץ ההגשה צריך להיות קובץ `zip` המכיל שני קבצים: `part1.c` ו-`mtm_buggy.c` (מתוקן).

- אין צורך לצרף קבצים שסופקו לכם, על קובץ ה-`zip` להכיל רק את קבצי ה-`C` שכתבתם בעצמכם.
- על הקובץ להיות מכוון כ-`zip` (לא `rar` או כל דבר אחר) כאשר קבצי הקוד נמצאים בתיקיה הראשית בקובץ ה-`zip`.

- סטודנטים המעוניינים לנצל ימי איחור, צריכים להגיש קובץ טקסט בודד תחת HW0_LATE המכיל מספר יחיד שבו רשומים מספר ימי האיחור אותם אתם מעוניינים לנצל.

7 שינויים עדכונים והודעות בנוגע לתרגיל

כל ההודעות הנוגעות בתרגיל ימצאו באתר של הקורס <http://webcourse.cs.technion.ac.il/234122> בדף התרגילים. דף זה יכיל שאלות ותשובות נפוצות. רק הודעות דחופות תשלחנה בדואר. עליכם לעקוב אחר האתר והעדכונים שיפורסמו בו.

בהצלחה !