

מכללה אקדמית הדסה החוג למדעי המחשב

הקורס: תכנות מודולארי א'

הנחיות לעבודה במערכת הפעלה Linux (באמצעות Xmanager)

את תכניות המחשב שתכתבו בקורס, תריצו על מערכת הפעלה Linux. בדפים אלה נסביר כיצד תעבדו במערכת הפעלה זאת, ונציג מספר פקודות בסיסיות להן תזדקקו לצורך העבודה. העבודה מול מערכת לינוקס נעשית בעיקרה על-ידי הקלדת הפקודה (באופנות עבודה טקסטואלית, בניגוד לעבודה עם ממשק גרפי כפי שעובדים מול תכנת חלונות). כמובן שעם הזמן כדאי (שלא לומר חובה) שתרחיבו את שליטתכם בלינוקס, מעבר לבסיס הראשוני אותו נלמד עתה.

לצערי, חלק ניר מהתלמידים בקורס משתדלים להתחמק מלהכיר את מערכת הפעלה לינוקס, על-ידי שהם עושים את מרבית העבודה בסביבת חלונות, ורק את המינימום ההכרחי הם מבצעים בלינוקס. זה חבל, שכן הכרות עם מ.ה. זאת חשובה למתכנת המכבד את עצמו, והמעוניין להרחיב את אופקיו וידיעותיו. כמו כן, כדי לא לסייע לכם 'לתקוע לעצכם גול עצמי' בכך שתתחמקו מלינוקס, תכלול הבחינה המסכמת בקורס מספר שאלות בנושא.

קראו כל סעיף בשלמותו ובהקפדה, טרם שאתם מבצעים את שיש לבצע באותו סעיף! הקפידו לראות שאתם גם מבינים למה ואיך הפקודה עושה את שהיא עושה, ומה תפקיד כל מרכיב בה. אל תהססו לשחק, לנסות לעשות דברים מעט שונים.

1. כדי להתחיל לעבוד מול מחשבי המכללה בהם מותקנת מערכת הפעלה Linux, מתוך המחשבים עליהם מותקנת מערכת הפעלה חלונות, עליכם להקיש קליק כפול על ה- icon של Xmanager (icon סגול בצורת X) המצוי על שולחן העבודה (ה- desktop) של תכנת חלונות. אם נפתח חלון השואל אתכם על חסימת התכנה הקישו על כפתור : Unblock.

2. ייפתח חלון חדש בשם Broadcast Chooser. בחלון זה הקישו קליק כפול על שם המחשב מולו אתם מעוניינים לעבוד (inferno-04, inferno-05, inferno-06), אחד מבין מחשבי המכללה המריצים Linux). אין מבחינתכם חשיבות מול איזה מחשב אתם עובדים (למעט אולי העומס על המחשבים השונים, עליו תוכלו ללמוד חלקית ממספר המשתמשים בכל מחשב, כפי שהחלון מראה לכם).

3. עת אתם עובדים מול מערכת הפעלה Linux המחשב שלכם הופך למסוף העובד מול מחשב מרכזי, עליו עובדים תלמידים רבים במקביל (בדרך כלל בלי להיות מודעים לכך שהם עובדים במקביל מול מחשב אחד, ולא כל אחד על המחשב שעליו הוא עובד. אנו אומרים כי העבודה מול המחשב מרכזי [server] הינה **שקופה** לכם במובן זה שאינכם 'רואים' אותה, אינכם חשים בה).

4. נפתח חלון. לפני שאתם פונים להזנת שם המשתמש שלכם פנו ללשונית Session והקליקו עליה. בחלון שנפתח קבעו את Xfce4 להיות מנהל שולחן העבודה

שלכם (ה- desktop manager) (במקום המחדל שהוא Gnome). בינתיים קבעתם את מנהל שולחן העבודה שלכם להיות XFce4 רק לפרק העבודה הנוכחי. בהמשך נראה כיצד קובעים אותו להיות מנהל שולחן העבודה באופן קבוע. (בלינוקס, בניגוד לחלונות, קיימות תוכנות שונות המנהלות את שולחן העבודה באמצעותו אתם עובדים מול המחשב, ליתר דיוק מול מערכת ההפעלה. XFce4 היא אחת התוכנות, והמומלצת ע"י מנהל המערכת במקומותינו).

5. עתה הזינו את ה- login שלכם **(במערכת Linux)**, שאינו זהה בהכרח לזה שבמערכת חלונות, אם כי בפועל במקומותינו הוא זהה, ויש לכך חשיבות! , **ואחר Enter**. בדרך כלל יהיה עליכם, אחרי הקשת ה- login, להקיש את הסיסמה (password) שלכם (ואחר Enter). בכניסה הראשונה למערכת, טרם שקבעתם לעצמכם סיסמה, לא תתבקשו להזין סיסמה. מייד נלמד כיצד לשנות את הסיסמה, בפרט כיצד לקבוע סיסמה בפעם הראשונה. חובה לשנות את הסיסמה במהלך עבודתכם היום על המחשב!!

6. לוקח למערכת Linux רגע לעלות. אנו עובדים עם הפצה (דיאלקט) הנקרא CentOS והינו גרסה חינוכית של Red Hat Linux. בתחתית המסך מופיע **פִּנֵּל** (במילים אחרות סרגל כלים) של מערכת לינוקס.

7. בפנל של XFCE הקישו על הכפתור השמאלי (Terminal), זה עם ציור המסך. בכך אתם פותחים חלון **Shell**, כלומר חלון המאפשר לכם להזין פקודות למחשב באופן טקסטואלי, רוצה לומר על-ידי כתיבת הפקודות כטקסט (בניגוד לעבודה בממשק גרפי כפי שנהוג במערכת חלונות). בהמשך נכיר פקודות שונות שאתם יכולים להזין כטקסט, כמובן. ניתן, ולעיתים גם נוח, לפתוח כמה חלונות **Shell**. (מקור המונח **shell**, כלומר צדף, הוא בכך שממשק זה הוא מעין צדף העוטף את מערכת ההפעלה, ובאמצעותו אתם פונים למערכת ההפעלה).

8. הפקודה הראשונה של מערכת הפעלה Linux שנכיר היא הפקודה לשינוי סיסמה (password). בחלון ה־**שֵׁל** כתבו את הפקודה **yppasswd**, ואחר **Enter**. המחשב יבקש מכם להזין את הסיסמה הישנה (אם הייתה כזאת), ופעמיים את החדשה (כדי לוודא שזאת הסיסמה לה התכוונתם) הסיסמה שאתם מקישים לא תראה על המסך, **גם לא ממוסכת בכוכביות**, אל תיבהלו מכך. הקפידו לזכור את הסיסמה שלכם! הסיסמה צריכה לכלול לכל הפחות שישה תווים, ולכל היותר שמונה, וכדאי שהסיסמה תכיל גם תווים שאינם אותיות או ספרות (למשל !, ?, *, או כל תו אחר). תלמיד שלא ישנה את סיסמתו לא יוכל להיכנס לעבוד (שכן אנשי המחשבים של המכללה יקבעו לו סיסמה שתהא, לפיכך, ידועה רק להם). **במערכת לינוקס יש אבחנה בין אות גדולה לקטנה, על כן את הפקודה yppasswd יש לכתוב באות קטנה בדיוק כמו שכתבנו כאן (ולא YPPASSWD או Yppasswd); כך גם לגבי כל יתר הפקודות שנתאר כאן. כמובן שיש להקפיד גם על רווחים, מקפים (-) וכל מה שכתוב בדפים.**

9. מכיוון שכדאי מאוד שסיסמתכם בלינוקס תהיה זהה לזאת שבחלונות (אחרת עלולות להיווצר בעיות, למשל, בהדפסה, או בהעברת קבצים בין המערכות) החליפו גם את סיסמתכם בחלונות. כדי לעשות זאת הקישו: **Ctrl + Alt + Del** ובחלון שנפתח בקשו להחליף סיסמה.

10. הפקודה השנייה של מערכת הפעלה Linux שנכיר היא הפקודה `date`. הקישו אותה, וודאו שהמחשב אכן מעודכן מבחינת התאריך והשעה.

11. הפקודה הבאה של מערכת הפעלה לינוקס שנכיר היא פקודת יצירת מדריך (במילים אחרות `directory` או מחיצה כלומר `folder`). הפקודה הינה `mkdir` (קיצור של `make directory`) ויש לספק לה את שם המדריך שברצוננו ליצור. על כן הקלידו `mkdir modprog1` (וכמובן, אחר `Enter`). בזאת יצרתם במדריך בו אתם מצויים עתה תת-מדריך בשם `modprog1` (קיצור של `modular programming`). המדריך `modprog1` ישמש אתכם עבור הקורס שלנו. כל החומר שתיצרו במסגרת הקורס ישכון במחיצה זאת, בפרט בתת-מחיצות שלה. כדי ליצור גם מדריך עבור הקורס באלגברה אתם יכולים לכתוב: `mkdir algebra`. המדריך בו אתם נמצאים עתה אתם נכנסים לעבודה מול המחשב נקרא **מדריך הבית** (`home directory`) שלכם, הוא מסומן כ- `~yosico` עבור תלמיד שהלוגין שלו הוא `yosico`.

12. כדי לעבור למדריך `modprog1`, שיצרנו זה עתה, הקישו את הפקודה: `cd modprog1` (קיצור של `change directory`). כדי לחזור למדריך הבית שלכם הקישו `cd ..` (ככלל, הארגומנט: `..` מציין את המדריך שמעל המדריך הנוכחי, ולכן אם אתם מבקשים לעבור למדריך `..`, אתם עוברים למדריך שמעל הנוכחי). אתם מוזמנים לטייל בין המדריכים במחשב לעלות ולרדת בהם להנאתכם. מכל מקום אליו הגעתם תוכלו לשוב ישירות למדריך הבית שלכם על-ידי שתקישו `cd ~yosico` (בהנחה שהלוגין שלכם הוא `yosico`). לחילופין אתם יכולים להקיש גם `cd ~` או `cd` גם שתי גרסות אלה של הפקודה `cd` תחזרנה אתכם למדריך הבית שלכם.

13. הפקודה `pwd` (קיצור של `present working directory`) מציגה לכם את המדריך בו אתם עובדים, והפקודה `whoami` מציגה את הלוגין שלכם.

14. חיזרו למדריך `modprog1` וצרו בו את מדריך `ex0` (באמצעות הפקודה `mkdir`) שהוא מדריך בו יישמרו כל הקבצים של תרגיל מספר אפס. היכנסו למדריך `ex0` (בעזרת הפקודה `cd`).

15. עתה נרצה לצור את קובץ המקור (`source`) שיכיל את התכנית שעלינו לכתוב במסגרת תרגיל מספר אפס. הקובץ ייקרא `ex0.cc`. כדי לצור את הקובץ נזדקק לעורך (`editor`). אנו נשתמש בעורך ששמו `emacs`. כדי להפעיל את העורך הקישו בחלון הפסל: `& emacs ex0.cc`. בזאת הורנו למחשב להפעיל את העורך `emacs`, על הקובץ (שבמקרה זה עדיין לא קיים) `ex0.cc`. בשלב זה לא נסביר מדוע יש לכתוב את סימן ה- `&` (אמפֶּרְסַנְט) בסוף הפקודה. מהו עורך? ומה ההבדל בינו לבין מעבד תמלילים? נבהיר שלא בדפים אלה.

16. נפתח חלון חדש ובו תוכלו להקליד את התכנית הרצויה. מכיוון שאנו עדיין לא למדנו לכתוב תכניות הקלידו את שמכם וכתובתכם. בהמשך, במקום זה תקלידו את תכניתכם.

17. התכנית שהקלדתם עד עתה מצויה בחוצץ (`buffer`), כלומר בזיכרון הראשי. כדי לשמור אותה כקובץ על-גבי הדיסק של המחשב בחרו בתפריט `Files` את השורה `Save (current buffer)`. בשלב זה גם נוכל לצאת מ- `emacs`: בתפריט `Files` בחרו

בשורה האחרונה: `Exit emacs`. (לצד כל שורה בתפריטים השונים מופיע האופן בו ניתן להזינה ישירות מהמקלדת, ללא שימוש בתפריטים, עם הזמן כדאי לכם ללמוד את הקיצורים הללו). אתם, כמובן, מוזמנים להקליד חלק מהתכנית, לשמרה, להקליד עוד חלק, ואחר לשמור גם אותו, כלומר לבצע מספר פעולות שמירה.

18. דלגו על סעיפים 17-21 (וחזרו אליהם אחרי שתלמדו לכתוב תכנית בשפת `C++`). נחזרו עתה לחלון ה-`Shell`. בחלון זה נרצה לקמפל את התכנית (כלומר לתרגמה לשפת מכונה). לשם כך עלינו להפעיל את תכנת הקומפיילר, וזאת נעשה באמצעות הפקודה: `g++ -Wall ex0.cc`. (שימו לב כי אנו כותבים `W` באות גדולה, ובצמוד לו `all` באות קטנה). נסביר את הפקודה: `g++` הוא שמו של הקומפיילר; הפסוקית: `-Wall` מציינת כי על הקומפיילר להקפיד בבדיקה התכנית, ולא לקבל 'קוד שנראה לו חשוד'; המרכיב האחרון בפקודה מציין את שמו של קובץ המקור שיש לקמפל.

19. קראו את הודעות השגיאה שהקומפיילר מודיע לכם. שימו לב כי עבור כל שגיאה מתוארת השורה בה הקומפיילר **איתר את השגיאה**, וכן תיאור של השגיאה. חיזרו לחלון העורך ותקנו את השגיאות. תנו דעתכם לכך שאם שכחתם נקודה-פסיק בסוף שורה מספר `x`, אזי הקומפיילר מגלה את חסרונה רק בשורה מספר `x+1`; באופן כללי, במקרים רבים השגיאה שלכם חלה מעט לפני המקום בו קומפיילר מאתר אותה. שימו לב כי בתחתית חלון העורך מופיע מספרה של השורה בה אתם מצויים, וכך נקל לאתר את השורה אליה עליכם לפנות. פקודת העורך `Alt g` מאפשרת התקדמות לשורה מבוקשת. אחרי התיקונים שימרו שוב את הקובץ.

20. חיזרו לחלון ה-`Shell` וקמפלו שוב את הקובץ (הקשה על מקש החץ כלפי מעלה תגרום לכך שהמחשב יציג את הפקודה הקודמת שהזנתם בחלון השל, אתם יכולים להיעזר בכך כדי לחסוך לכם את הצורך להקליד את הפקודה שוב. כמו כן, אם תקישו חלק ראשון, במילים אחרות רישא, של שם פקודה, ואחר תקישו על מקש ה-`tab` ישלים עבורכם המחשב את שם הפקודה. נסו). אם תיקנתם את תכניתכם כהלכה אזי לא תתקבל כל הודעת שגיאה, והקומפיילר (יחד עם בן זוגו הלינקר) יתרגמו את התכנית שלכם לשפת מכונה, תוך שהם יוצרים את הקובץ `a.out` אשר מכיל את תרגום התכנית לשפת מכונה (במילים אחרות את ה-`executable code` של התכנית, אשר נקרא לעיתים, על-ידי מי שאינם מדייקים, גם ה-`object code` של התכנית).

21. כדי להריץ את התכנית הקישו בחלון השל: `a.out`.

22. אתם מוזמנים לשוב לתכניתכם, להוסיף לה עוד פקודה או שתיים, לשמרה שוב, לקמפלה שוב, ולהריצה שוב.

עתה נרצה להכיר עוד מספר פקודות אותן ניתן להורות למערכת ההפעלה, באמצעות ה-`Shell`:

23. כדי להציג על המסך תוכנו של קובץ נשתמש בפקודה `less`. לדוגמה: `less ex0.cc` יציג את תוכנו של הקובץ `ex0.cc` על-גבי המסך. (הפקודה `less` הינה חלופית ל-`more`, זוהי פקודה מתוחכמת יותר). במידה והקובץ משתרע על-פני יותר ממסך אחד יוצג רק חלקו הראשון. הקשה על `Enter` תגרום להצגת שורה בודדת נוספת, הקשה על מקש הרווח תגרום להצגת חלקו הבא של הקובץ. עת סיימתם לצפות בקובץ הקישו על המקש `q` (בשביל `quit`), וחזרו לעבודה מול ה-`Shell`.

24. כדי לראות אילו קבצים מצויים במדריך בו אנו מצויים עתה נקיש את הפקודה: `ls` (קיצור של `list directory`). אם תקישו אותה תגלו שבמדריך הנוכחי מצוי

הקובץ : ex0.cc. (אלה מכם ששמרו את הקובץ ex0.cc מספר פעמים, אחרי שהם בצעו מספר שנויים בקובץ, יגלו כי במדריך קיים גם הקובץ ex0.cc~. קובץ זה נוצר על-ידי emacs, הוא מכיל את הגרסה של הקובץ טרם עדכונו האחרון). הפקודה : ls -l תציג מידע רב יותר אודות כל קובץ, למשל את גודלו, ומתי הוא שונה לאחרונה. (שימו לב כי הדגל כולל את האות הלטינית el, ולא את הספרה 1). להזכירכם : l - הוא דגל של הפקודה המוכה למחשב כי יש להציג את המידע אודות הקבצים הרצויים בפורמט ארוך (l = long). אם אינכם כותבים את הדגל מבין המחשב כי די לכם בפורמט הקצר. אנו אומרים כי **המחדל** (ה- default) הוא הפורמט הקצר : הוא זה שמבוצע אם לא נאמר אחרת. בדקו לעצמכם, מתישהו, מה עושים הדגלים a - ו- R -.

25. הפקודה cp (קיצור של copy) מעתיקה את תוכנו של קובץ אחד על קובץ שני, ובכך משכפלת את הקובץ. לדוגמה : cp ex0.cc xxx תעתיק את הקובץ ex0.cc על הקובץ xxx. במידה וקובץ בשם xxx לא היה קיים עד כה הוא יִוצר עתה, במידה ובעבר היה קובץ בשם אזי תוכנו הישן יאבד ויוחלף בתוכנו של ex0.cc. כדי להעתיק קובץ הנמצא במדריך אחר למדריך הנוכחי, כך שהוא ייקרא במדריך הנוכחי באותו שם בו הוא קרוי במדריך ממנו הוא מועתק כתבו לדוגמה : cp /usr/include/stdio.h (הנקודה בסוף הפקודה מציינת כי יש להעתיק למקום הנוכחי, ולשמור על אותו שם שיש לו). כך אתם מעתיקים את הקובץ stdio.h הנמצא במדריך /usr/include למדריך בו אתם נמצאים עתה (ואשר מצוין על-ידי התו נקודה המופיעה מוצנעת בסוף הפקודה!). לבדיקתכם : תפקיד הדגל : -r.

26. הפקודה mv (קיצור של move) משנה שם של קובץ או מעבירה אותו למדריך אחר. לדוגמה הקלידו : mv xxx yyy ועל ידי כך שנו את שמו של הקובץ xxx לשם החדש yyy. עתה הקלידו mv yyy .. ובכך תעבירו את הקובץ yyy למדריך שמעל המדריך הנוכחי, במילים אחרות למדריך modprog1. (להזכירכם : .. מתייחס תמיד למדריך מעל, כפי שנקודה בודדת מתייחסת למדריך הנוכחי).

27. הפקודה rm (קיצור של remove) מוחקת קובץ. למשל מחקו את הקובץ yyy על-ידי שתתקדמו למדריך modprog1 (באמצעות : cd ..) ואחר rm yyy. עתה ודאו באמצעות הפקודה ls כי הקובץ אכן נמחק. לבדיקתכם : תפקיד הדגל : -r.

28. הפקודה pwd (קיצור של present working directory) מדפיסה את שמו של המדריך בו אתם מצויים עתה.

29. כדי למחוק מדריך השתמשו בפקודה rmdir. למשל חיזרו למדריך הבית שלכם (כיצד?) ומחקו את המדריך algebra.

30. הפקודה באמצעותה ניתן להדפיס קובץ היא הפקודה : a2ps. יש לציין בפקודה את שם הקובץ שברצוננו להדפיס, באיזה מדפסת ברצוננו להשתמש, ומה שם המשתמש (במערכת חלונות). לדוגמה, נוכל לכתוב : a2ps ex0.cc labpr2 yosico בכך אנו מבקשים להדפיס את הקובץ ex0.cc במדפסת labpr2 (המצויה בסמוך למעבדה #2). המשתמש המדפיס הוא yosico. אחר תתבקשו גם להזין את סיסמתכם בחלונות, ואז בהנחה שתקציב ההדפסה שלכם אינו ריק, הקובץ יודפס, (וחשבונכם, כמוכן, יחוייב).

31. הפקודה : XFCE switchdesk תעזור לכם לשנות את מנהל שולחן העבודה שלכם להיות XFCE באופן קבוע.

32. הפקודה man (קיצור של manual) מציגה הסבר אודות פקודות שונות של מערכת ההפעלה, או של שפת C. למשל נסו לכתוב : man ls או man whatis.

33. כדי לסיים את העבודה בחלון Shell הזינו את הפקודה exit.

34. כדי לסיים את העבודה ב-linux לחצו על הכפתור הימני (Exit) בפנל שבתחתית המסך. **הקפידו לצאת מלינוקס בצורה מסודרת**, ולא רק על-ידי כיבוי המחשב.

זה יחסוך לכם צרות! במידה ובכך סיימתם גם לעבוד על המחשב, וברצונכם לעזוב את המעבדה, זכרו גם לצאת מהחשבון שלכם בחלונות.

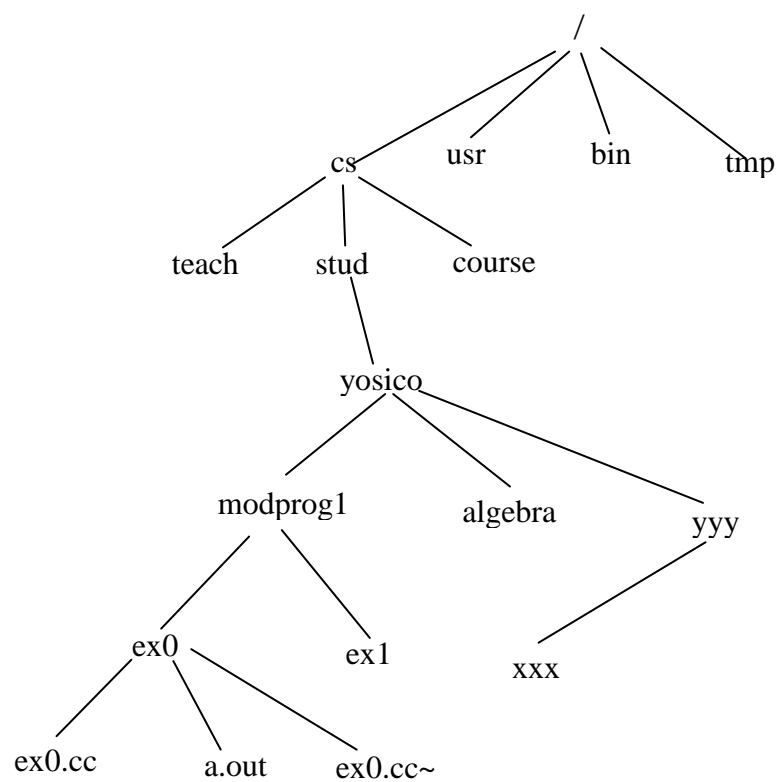
35. כדי להעתיק קבצים שיצרתם בלינוקס למערכת חלונות בצעו את השלבים הבאים: בחלונות, הקישו על כפתור Start, ובקשו את האופציה Run... בחלון שנפתח, אתם יכולים לבקש להריץ פקודה, לכן הזינו את הפקודה: <http://linuxfs.yosico> (בהנחה ששם המשתמש שלכם הוא yosico). ייפתח חלון ובו מדריך הבית שלכם בלינוקס, ממנו תוכלו להעתיק קבצים כאוות נפשכם. הערה: כדי ששיטה זאת תפעל צריך שהסיסמה שלכם בחלונות ובלינוקס תהיה זהה! לחילופין: פתחו את ה- File Explorer (קליק ימני על כפתור Start, ובקשו Explore). בתפריט Tools בחרו: Map Network Drive... בחלון שנפתח בקשו בתיבה העליונה למפות את כונן Y:, ובתיבה התחתונה למפות ולקשרו ל: <http://linuxfs.yosico>. עתה פניה לכונן Y: תביאכם למדריך הבית שלכם בחלונות. לחילופין: בעזרת דפדפן (כדוגמת Internet Explorer) פנו לכתובת: <ftp://yosico@10.2.10.24>, ומשם המשיכו בכוחות עצמכם.

מבנה מערכת הקבצים

עת אתם נכנסים לעבוד מול? מערכת לינוקס (כלומר אתם עושים login), אתם, כאמור, נכנסים למדריך הבית (home directory) שלכם. היכן ממוקם אותו מדריך בית במערכת הקבצים של המחשב? וכיצד נראית מערכת הקבצים באופן כללי? מערכת הקבצים בלינוקס בנויה כמין עץ (הפוך) המורכב משורש, שהולך ומתפצל (כמו שענפי העץ הגדל ביער הולכים ומתפצלים). שורש מערכת הקבצים מסומן כ- / ונקרא root directory (שימו לב כי הלכנס במערכת לינוקס נוטה ימינה, בניגוד לזה של חלונות שנוטה שמאלה). לשורש ישנם כמה מדריכי משנה, למשל המדריך cs מכיל חומר הקשור למדעי המחשב (cs עומד בשביל Computer Science). המדריך bin (קיצור של binary) מכיל תכניות שונות. המדריך usr מכיל גם הוא תכניות וקבצים שונים אשר משמשים משתמשים (users) רבים. כל אחד ממדריכי המשנה של השורש מסומן כ: /cs או /usr וכו'. נתמקד עתה במדריך /cs. גם הוא מחולק למספר מדריכים. תת-המדריך teach מכיל חומר של מורים שונים, תת-המדריך stud מכיל את מדריכי הבית שלכם. ועל-כן מדריך הבית של התלמיד שהתקציב שלו הוא yosico הינו ליתר דיוק: /cs/stud/yosico. תת-מדריך שלישי של המדריך /cs הוא course. תת-מדריך זה מכיל מידע אודות קורסים שונים, כדוגמת הקורס שלנו. על-כן המדריך: /cs/course/modprog1 הוא זה המכיל חומר הקשור לקורס שלנו. חלק מחומר זה ישמש אתכם לאורך הקורס. תת-מדריך נוסף של השורש הוא /tmp ועליו כדאי שתלמדו בהזדמנות.

עתה אתם מוזמנים לשוטט במעלה ובמורד מערכת הקבצים (להזכירכם .. cd מעלה אתכם רמה אחת כלפי מעלה בעץ, ו- cd מחזיר אתכם למדריך הבית שלכם).

נוכל לצייר את מערכת הקבצים באופן הבא (זהו ציור חלקי):



מכאן ואילך אציג נושאים אותם אינכם אמורים ללמוד כבר היום; אך כן מצופים ללמוד במהלך הסמסטר, אחרי שתשתלטו על החומר הבסיסי יותר הקשור לתפעול המחשב, ואשר הוצג מעל. אדגיש ואגיד שמי שהחומר שנלמד עד כה מאיים להציף אותו, עדיף שיניח לעת עתה את יתר ההסברים; **אך יחזור אליהם בהמשך!** מנגד, על כל פקודה אני מציג כאן רק את השימוש הבסיסי ביותר, מומלץ לרכוש את רזי השימושים היותר שלמים בפקודות הללו.

הרשאות (permissions)

הרשאה היא תכונה של קובץ המציינת מי רשאי: (א) לשנות את הקובץ (לכתוב עליו), (ב) לקרוא את הקובץ (המצוי אי-שם במחשב, ולכן לכאורה, עלול 'להידרס' על-ידי כל משתמש מרושע המשתמש אתכם באותו מחשב), (ג) להריץ את הקובץ (בהנחה שהוא כולל תכנית הניתנת להרצה). למשל, את קובץ הבחינה בקורס סביר להניח שרק המורה רשאי לכתוב, ורק המורה רשאי לקרוא, ואיש אינו רשאי להריץ (זו אינה תכנית), לעומת זאת את הקובץ המתאר את תרגיל מספר 12 רשאי רק המורה לשנות, אך כל תלמיד רשאי לקרוא (כדי לדעת מה על אותו תלמיד לעשות בתרגיל מספר 12). במערכת Linux מבחינים בין: (א) הבעלים של הקובץ (user או בקיצור u), שיצר את הקובץ, לבין (ב) משתמשים המוגדרים כ-'קבוצה' של אותו משתמש (עבורכם אלה התלמידים האחרים, group או בקיצור g), לבין (ג) כלל המשתמשים במחשב (other או בקיצור o). עבור כל אחת משלוש הקבוצות הללו (משתמש, קבוצה, הכלל) יש להגדיר האם מי שנמצא באותה קבוצה רשאי לשנות/לכתוב את הקובץ? האם מי שנמצא באותה קבוצה רשאי לקרוא את הקובץ? האם מי שנמצא באותה קבוצה רשאי להריץ את הקובץ. למשל, קובץ המבחן יוגדר ככזה שהבעלים שלו רשאי לקרוא ולכתוב (אך לא להריצו, שכן הוא אינו תכנית); הקבוצה של הבעלים אינה רשאית לא לקרוא ולא לכתוב; כלל המשתמשים לא רשאים לא לקרוא ולא לכתוב. כיצד יוכל המורה לדאוג להשיג תוצאה רצויה זאת? באמצעות הפקודה `chmod` (קיצור של `change mode`). ניתן לכתוב את הפקודה בדרכים שונות; אציג כאן את הצורה הפשוטה ביותר. למשל נוכל לכתוב: `chmod go-r exam` בזאת שינינו את ההרשאה הקיימת כך ש: הקבוצה (g) של המשתמש, ויתר המשתמשים (o) אינם יכולים לקרוא: `-r` את הקובץ `exam` (התו מינוס אומר שהאפשרות נלקחת מאותה אוכלוסיה, בניגוד לתו פלוס שאומר שההרשאה ניתנת לאותם משתמשים). באופן דומה אם נכתוב: `chmod g+w some-file` אזי חברי הקבוצה (g) של המשתמש שביצע את הפקודה יהיו רשאים לכתוב (+w) על הקובץ `some-file`. `chmod u+x some-file` מאפשר לבעלים להריץ את הקובץ. (במידה ונקמפל תוכנית הקובץ שיווצר באופן אוטומטי יהיה קובץ הרצה). במידה ובידנו קובץ נתון, וברצוננו לדעת מהן ההרשאות שלו ניתן להשתמש בפקודה `ls -l some-file` (הדגל `-l` כולל את האות האנגלית `l`, כדי לציין פורמט ארוך של פקודת ה-`ls`). בפלט שיתקבל התו הראשון בשורה מציין את סוג הקובץ: - מורה שמדובר בקובץ רגיל, `d` מורה שמדובר במדריך/מחיצה (וקיימות אפשרויות נוספות שלא אמנה כאן). תשעת התווים שאחר כך כוללים מקפים, ואותיות מתוך הקבוצה {r,w,x}: שלושת התווים השני, שלישי, רביעי בשורה מתייחסים לבעלים של הקובץ. התווים חמישי, שישי, שביעי, מתייחסים לקבוצה, התווים שמיני, תשיעי, עשירי, מתייחסים לכלל המשתמשים. בכל שלשה כזאת התו השמאלי ביותר מתייחס לפעולת הקריאה, אם ערכו הוא `r` אזי המשתמש(ים) המתאים רשאי(ים) לקרוא את הקובץ, אם ערכו הוא - אזי המשתמש(ים) אינו(ם) רשאים לקרוא. התו האמצעי מתייחס לכתיבה, והוא עשוי להיות `w` אם יש אפשרות כתיבה, או - אם אין. התו הימני מתייחס להרצה, והוא עשוי להיות `x` אם יש אפשרות הרצה, או - אם אין. לדוגמה: `-r-x--x-w-` משמעו: מדובר בקובץ רגיל (התו הראשון הוא -),

הבעלים רשאי לקראו ולהריצו (בשל התווים: $r-x$ במקומות השני עד הרביעי, והחלפת התו w ב: $-$), הקבוצה שלו רשאית רק להריצו ($--x$ מורה על כך), כלל המשתמשים רשאים רק לכתוב עליו ($-w$ מורה על כך).

פקודות נוספות שכדאי להכיר

1. **date** פקודה זאת מציגה את התאריך והשעה הנוכחיים.
2. **history** פקודה זאת מציגה בפניכם את הפקודות שהקשטם בעבר. לצד כל פקודה מוצג מספרה הסידורי. כדי לחזור ולהריץ את הפקודת שמספרה, לדוגמה, 17, אתם יכולים להקיש 17.

כדי שכל הפקודות הרבות שהצגתם בעבר לא יחלפו על פניכם ביעף אתם יכולים להקיש: **history | less** (בין שתי הפקודות הפרדנו בקו אנכי, המצוי בד"כ על המקלדת ליד מקש ה-Enter, ומצויר כקו אנכי המחולק לשנים). מה למעשה עשיתם? ביצעתם פעולה הנקראת pipe (מלשון צינור): את הפלט של פקודת ה-history שלחתם כקלט לפקודת ה-less, אשר כדרכה, מציגה את הנתונים מסך, מסך (והקשה על רווח גורמת לה להתקדם ממסך למסך; q גורם ליציאה).

3. **grep** פקודה זאת מציגה לכם שורות בקובץ (או במספר קבצים), המכילות מילה אותה אתם מחפשים. לדוגמה: **grep "Hello" file1 file2 file3** יציג בפניכם שורות מתוך הקבצים file1, file2, file3 אשר מכילות את המילה Hello. באופן דומה: **history | grep "a2ps"** יבצע את המהלך הבא: תורץ פקודת ה-history, הפלט שלה יישלח כקלט לפקודת ה-grep "a2ps" אשר תציג בפנינו את הפקודות שביצענו בעבר, ואשר כללו בתוכן את המילה\מחרוזת a2ps. בדקו את תפקיד הדגלים הבאים של פקודת ה-grep:
-i -l -w .

4.

diff פקודה זאת מסייעת לנו להשוות בין קבצים (דומים), ולאתר את ההבדלים שקיימים ביניהם. לדוגמה, צרו זוג קבצים (זהים):

<u>file1</u> 111 222 333 444 555	<u>file2</u> 111 222 333 444 555
---	---

והריצו: diff file1 file2

בין הקבצים לא נמצא כל הבדל, ועל כן הפלט נותר ריק.
עתה שנו את file1 בלבד להיות:

<u>file1</u> 111 222 33x3 444 555
--

והריצו שוב את פקודת ה-diff. מתקבל הפלט:

2c2 < 33x3 --- > 333

5.

נסביר:

עתה קיים הבדל בין שורה #2 בקובץ השמאלי, לשורה #2 בימני, והמרכיב 2c2 מעיד על כך. כמו כן, עם קו המפריד ביניהן (---), מוצגת השורה מהקובץ הראשון כשלצדה <, והשורה, השונה ממנה, מהקובץ השני כשלצידה >

<u>file1</u> 111 222 33x3 444 555
--

עתה שנו את הקובץ הראשון להיות:
(כלומר הוסיפו שורה ריקה, מתחת ל: 33x3)

והריצו שוב הפקודה: הפלט עתה אומר לנו כי שורות 2,3 בשמאלי ו: 2 בימני שונות, וכמו קודם מציג את ההבדל עם קו בין הקובץ הראשון לשני.

עתה הוסיפו לקובץ file1 שורה חדשה, בסוף הקובץ, הכוללת את 666 והריצו שוב diff file1 file2. הפעם הפלט מורה לנו שקיימים שני הבדלים: הראשון כמו קודם, והשני בין שורות 4 ו-5 בהן בקובץ השמאלי מופיעה המחרוזת 666, ובימני כלום.
(שאלונת: בדקו, בעזרת דפי ה-man מה תפקיד הדגלים: -b ו- -B)

6. **file** פקודה זאת מציגה לכם מידע אודות מהות המידע השמור בקובץ שהועבר לה כארגומנט: האם מדובר בקובץ טקסט, בקובץ ניתן להרצה, במדריך, ועוד. נסו למשל: `file ex0.cc` ולעומת זאת: `file a.out` או `ex0`.
7. **locate** ו-**find**. **locate** היא פקודה העוזרת לכם לאתר היכן מצוי קובץ רצוי במערכת הקבצים במחשב המקומי (מדריכי הבית שלכם, לדוגמה, אינם מצויים במכונה זאת). נסו לדוגמה: `locate iostream`. הפקודה מסתמכת על מידע שנאסף כל לילה, על-כן לא תאתר קובץ שיוצר היום. חלופית לה היא פקודת ה-**find**. יש להעביר לה את שם המחיצה ממנה יש להתחיל בחיפוש, אחר את הדגל `-name`, ואחר את שם הקובץ הרצוי. לדוגמה: `find ~yoramb -name my_cmp.c` יאתר בתוך המחיצה `~yoramb` (וכל תת המחיצות שלה) את הקובץ `my_cmp.c`.
8. **sort** היא פקודה למיון שורות קובץ. לדוגמה: `ls /cs/stud | sort | less`. פקודת ה: `ls /cs/stud` מציגה את שמות כלל החשבונות של כלל התלמידים. כדי לקבל את הרשימה ממויינת נשלח את הפלט שלה להיות הקלט של `sort` אשר תמין את החשבונות. מכיוון שבמערכת יש חשבונות רבים, אזי כדי להציגם בצורה נוחה, בלי שהם יטוּסו לנו על-פני המסך, נשלח את הפלט של `sort` להיות הקלט של `less`.
9. **cat** היא פקודה המציגה קובץ, או סדרת קבצים, בזה אחר זה על המסך. אפשר לראות בה גרסה פרימיטיבית של `less`. **head** מציגה רק את עשר שורותיו הראשונות של הקובץ; בהתאמה **tail** מציגה רק את העשר האחרונות (`tail -20 ex0.cc` תציג את 20 השורות האחרונות, `tail +20 ex0.cc` תציג את כל השורות החל בשורה מספר עשרים).
10. **wc** (`wc = word count`). מונה את מספר התווים, מילים ושורות בקובץ.
11. **cut** מציגה רק חלק מכל שורה. לדוגמה: `cut -c1-3 data.txt` תציג את שלושת התווים הראשונים (c= characters, 1-3 = first to third characters) מכל שורה מהקובץ `data.txt`.
12. **du** (`du = disk usage`) מציגה את שטח הדיסק שקבצים ומחיצות תופסים (ביחידות של kb). כדי לדעת מה שטח הדיסק הכולל שנתפס על-ידכם בצעו: `du ~yosico | sort -n` הסבר: הפלט של `du` נשלח לפקודת המיון אשר ממיינת לפי הגודל המספרי, ועל כן השטח הכולל הנדרש על-ידכם מופיע אחרון. הקפידו שהשטח הנצרך על-ידכם לא יעלה על מאה מגה-בייט.
13. **jobs** מעניין מה עושה פקודה זאת? אולי היא עוזרת למצוא עבודה?
14. **kill** ועוד יותר מסקרן לדעת מה עושה זאת...
15. **man** (קיצור של manual) היא הפקודה באמצעותה נוכל ללמוד על פקודות `shell` שונות, כמו גם על פונקציות שונות בשפת C (אך לא ב: C++). הפקודה שולפת לנו מעין דף מידע מתוך סדרת 'ספרי ההסבר' המצויים במערכת. 'ספרי ההסבר' הללו מחולקים לתשעה 'כרכים' כאשר כל אחד מה-'כרכים' דן בנושא אחר (לדוגמה: כרך #1 דן בפקודות `shell`, פרק #3 בפונקציות ספריה של שפת C). אם תקישו: `man who` אזי בשורה הראשונה, בתחילתה ובסופה, בתוך סוגריים, יופיע מספר ה-'כרך' ממנו נלקח ההסבר המוצג לכם. לעיתים פקודה כלשהי תופיע בכמה 'כרכים' ואז, אלא אם תצינו אחרת, יופיע ההסבר מה-'כרך' הראשון בו מופיע הסבר אודות הפקודה. לדוגמה: פקודת ה- `kill` מוסברת הן בכרך #1 (ובו מתוארת פקודת ה- `shell` ששמה `kill`), והן בכרך #2 (בו מוסברת קריאת המערכת `kill`). כדי לבקש את ההסבר המופיע בכרך השני יש להקיש: `kill`

פקודת ה- man מציגה לנו מספר את המידע אודות הפקודה במספר סעיפים:

- א. name = שם הפקודה, ותיאור קצר שלה.
- ב. synopsis = תחביר הפקודה, האופנים השונים להשתמש בפקודה (בפרט דגלים אופציונליים של הפקודה)
- ג. description = הסבר מפורט יותר אודות הפקודה.
- ד. options = תיאור הדגלים השונים עימם ניתן להפעיל את הפקודה.
- ה. example = כשמה כן היא (לא תמיד תופיע).
- ו. see also = פקודות קשורות/דומות.
- ז. סעיפים נוספים שלא אזכיר כאן.

הפקודה : man -k something תציג לכם את באילו 'כרכים' מוזכרת המונח something. למשל : nsqrt -k .man

16. **whatis** נותנת הסבר קצר מהי פקודה כלשהי. נסו למשל : **whatis wc** . (בדקו גם את פקודות ה : **apropos**).
17. **finger** מציגה לכם מידע אודות משתמש רצוי כלשהו במחשב. נסו למשל : **finger yosico**.
18. **cal** (קיצור של calander) מציגה לוח שנה. נסו למשל להקיש : **cal 2000** או **cal 11 1947**.
19. **unix2dos** ו- **dos2unix** הן שתי פקודות חשובות המאפשרות לכם להעביר קבצים מפורמט דוס לפורמט יוניקס ולהפך. הסיבה לצורך בפקודה היא שבדוס סוף שורה מסומן באמצעות שני תווים : LF ו- CR (התווים מספר 10 ומספר 13 ב- ASCII); בעוד בלינוקס משתמשים רק בראשון ביניהם. לעתים, ההבדל הזה חסר משמעות, ואז גם אם לא תשתמשו בפקודה הכל יהיה בסדר, אך לעתים יש לו משמעות הרת גורל.
20. **touch** תשמש תאכם כדי 'לגעת' בקובץ ולעדכן את התאריך האחרון בו הקובץ שומש להיות התאריך הנוכחי.

טיפים

1. נסו את מקש החץ כלפי מעלה, הוא מחזיר אתכם אחורה בין הפקודות, וכך חוסך לכם את הצורך להקליד בשנית פקודה שכבר הקלדתם בעבר.
2. אם תקישו רישא (תחילית) של שם פקודה, שם מדריך או שם קובץ, ואחר על מקש ה-tab ישלים לכם המחשב את שם הפקודה/מדריך/קובץ בעצמו. כך תוכלו להיות בטוחים שהקלדתם את השם כהלכה.
3. CTRL+L הינו קיצור לניקוי המסך. (נסו!) CTRL+K מוחק שורה, CTRL+U מעתיק שורה, CTRL+A מחזיר סמן לראש השורה, CTRL+E מעביר סמן לסוף השורה.
4. some! יפעיל שוב את הפקודה הקודמת שהתחילה באותיות some
5. כדי להעתיק טקסט ממקום למקום: א. סמנו את הטקסט. ב. פנו למקום הרצוי, ושם לחצו על הכפתור העגול שבמרכז העכבר; הטקסט שסימנתם יישפך במקום הרצוי.
6. cd – מחזיר אותכם למדריך הקודם בו היתם. cd בלבד מחזיר אותכם למדריך הבית. נסו גם ~.cd
7. נניח שברצוננו לראות את כל הקבצים ששםם מסתיים ב: cc המצויים במדריך הנוכחי. נוכל להקיש: ls *cc הכוכבית מתאימה למספר כלשהו של תווים כלשהם. באופן דומה: ls ex3?.cc יציג את כל הקבצים ששםם מתחיל בשלוש האותיות ex3, אח"כ ניצב תו בודד כלשהו, ואח"כ שלושת התווים cc. ניתן להשתמש בסימני שאלה וכוכביות ככל העולה על רוחכם בפקודות שונות (לדוגמה: rm * יבקש למחוק את כל הקבצים במדריך הנוכחי). הנושא בכללותו, אליו פתחתי לכם רק צוהר קטן, כאן, נקרא **ביטויים רגולריים**, תפגשו אותו במקומות שונים ומגוונים, ובעזרתו תוכלו לשלוף/להציג/להתאים מחרוזות שונות ומגוונות בנוחות. אתן עוד כמה דוגמות לביטויים רגולריים: התו ^ מתייחס לתחילת שורה, על כן: grep "^my" my_file ישלוף שורות המתחילות במילה my. באופן דומה \$ מתייחס לסוף שורה; על כן: grep "bye\$" my_file ישלוף שורות הנגמרות במילה bye. הפקודה: grep "[A-Z]" my_file תשלוף שורות הכוללות אות גדולה בתחום שבין A ל-Z; הפקודה: grep "[^A-Z]" my_file תשלוף שורות שאינן כוללות אות גדולה; והפקודה: grep "[A-Z]" my_file תשלוף שורות המתחילות באות גדולה (שימו לב להבדל במיקום הכובע בין שתי הפקודות). הפקודה: grep "[0-9].." my_file תשלוף שורות הכוללות ספרה (בתחום אפס עד תשע), אח"כ זוג תווים כלשהם (נקודה מתאימה לכל תו בודד שהוא), ואחריהם אות גדולה.