

## תרגיל בית 1

**ניתן להגיש עד 30.04.2023 עד השעה 23:59**

**נא לקרוא את כל ההוראות לפני תחילת העבודה!!!**

בתרגיל זה עליכם לכתוב מספר תכניות בשפת C שרצות תחת (centos 7) UNIX כפתרון לתרגיל.

יש להגיש קובץ zip בשם hw1 המכיל את הקבצים הבאים:

1. קובץ Locked\_shell.c עם קוד התוכנית הראשי מתועד.
2. קובץ Std\_shell.c עם קוד התוכנית מתועד.
3. קובץ Lec\_shell.c עם קוד התוכנית מתועד.
4. קובץ WORD עם שמות הסטודנטים וצילום מסך להרצת התכנית.
5. קבצי C של הפקודות המיוחדות (כל מה שמסומן בירוק).
6. קובץ סקריפט להרצת ה shell : Sys\_shell.sh. שימו לב, בקובץ הסקריפט יכתבו פקודות קימפול והרצה בלבד. (כולל קימפול כל קבצי ה C שכתבתם)

מטרת התרגיל : שימוש בקריאות מ"ה של לינוקס

עליכם לממש shell חדש משלכם, ה shell יתמוך בכל הפקודות המפורטות בהמשך. חלק מהפקודות יעבדו ללא התחברות וחלק אחר רק אחרי ביצוע login. (ראו פירוט בהמשך).

- על ה shell להציג הודעת (prompt) בצורה הבאה :

LockedShell>

אחרי התחברות סטודנט:

StdShell>

אחרי התחברות מרצה:

LecShell>

על ה shell לתמוך בסדרת הפקודות הבאות שתפקידם העיקרי הפעלת עמדת מידע במכללה המאפשרת הצגת מערכת, יצירה מערכת ועוד פעולות המפורטות בהמשך.

- בעת הרצת ה shell, נוצרת תיקיה בשם Courses, ויוצרת תיקיה בשם student. המערכת עוברת על הקובץ std\_pass ובתוך תיקיה student יוצרת לכל סטודנט תיקיה לפי ID שלו.
- הערה – קובץ std\_pass נוצר מחוץ למערכת ומניחים שהוא קיים בעת הרצת ה shell. עליכם ליצור אחד באופן ידני. (ראו דוגמה למבנה ותוכן בסוף הקובץ).

### פעולות של ה Locked\_Shell :

#### : LoginStudent

פקודה זו מקבלת 2 פרמטרים : שם משתמש של סטודנט, סיסמה. בודקת נכונות הפרטים בעזרת המידע השמור בקובץ std\_Pass.txt. אם הנתונים נכונים מפעילה מריצה את Std\_Shell. במקרה וסטודנט הכניס סיסמה לא נכונה 3 פעמים המערכת חוסמת את הגישה שלו. גם אם יצליח לכנס אחר כך לא יצליח להריץ כל הפקודות. על מנת לאשר אותו מחדש עליו לפנות למרצה מחוץ למערכת והמרצה יחזיר לו את הגישה המלאה. (ראו דוגמה לקובץ std\_Pass בסוף הקובץ).

#### : LoginLecturer

פקודה זו מקבלת 2 פרמטרים : שם משתמש של מרצה, סיסמה. בודקת נכונות הפרטים בעזרת המידע השמור בקובץ lec\_pass. אם הנתונים נכונים מריצה את Lec\_Shell. (ראו דוגמה לקובץ lec\_Pass בסוף הקובץ).

#### : exit

פקודה זו מדפיסה הודעת **Goodbye** ויוצאת מהתוכנית, סוגרת את ה LockedShell וחוזרת ל- shell המקורי.

## פעולות של ה Std\_Shell:

רצה בתוך תיקיית student ותומכת בפקודות הבאות:

**ShowCourses**: (מציגה קורסים זמינים)

פקודה זו לא מקבלת פרמטרים.  
מציגה רשימת קורסים זמינים לרישום. (תוכן הקובץ Courses.txt שנמצא בתיקיית Courses .

**GetNewCourse**: (מציגה קורס אחרון שנוסף)

פקודה זו לא מקבלת פרמטרים.  
מציגה קורס אחרון שנוסף לקובץ Courses.txt שנמצא בתיקיית Courses .

**MakeSchedule**: (בניית מערכת שעות)

פקודה זו אינה מקבלת פרמטרים.  
פקודה זו נכנסת לתיקיית הסטודנט ובודקת במקרה ולא קיימת מערכת שעות:  
יוצרת קובץ חדש בשם Schedule.txt, מבקשת מהסטודנט להכניס קוד קורס.  
מציגה את לוח הזמנים של הקורס המבוקש ומבקשת ממנו לאשר על ידי הכנסת המילה Confirm או קוד קורס אחר.  
הכנסת Confirm תרשום את הקורס הרצוי עם הזמנים שלו בקובץ Schedule.txt,  
הקלדת Exit לסיום.  
במקרה והסטודנט לא אשר לפחות קורס אחד המערכת תמחק את קובץ Schedule.txt.

**GetSchedule**: (הצגת מערכת שעות)

פקודה זו אינה מקבלת פרמטרים.  
מדפיסה מערכת השעות של הסטודנט על המסך. במקרה ועדיין לא קיים מערכת שעות תוצג הודעה מתאימה. (ראו פלט לדוגמה).

**Logout**:

פקודה זו סוגרת את Std\_Shell, וחוזרת ל-Locked\_Shell.

## פעולות של ה Lec\_Shell:

**SetCourses** : (יצירת קורסים זמנים לרישום)

פקודה זו מקבלת 5 פרמטרים: קוד קורס, שם קורס, יום בשבוע, שעה התחלה ושעת סיום.  
פותחת קובץ בשם Courses.txt בתוך התיקייה Courses. מוסיפה את הקורס שהתקבל לסוף הקובץ.

**GetSchedule** : (הצגת מערכת שעות)

פקודה זו מקבלת פרמטר אחד: ID של סטודנט.  
מדפיסה מערכת השעות של הסטודנט על המסך. במקרה ועדיין לא קיים מערכת שעות תוצג הודעה מתאימה. (ראו פלט לדוגמה).

**DeleteStudent** : (מחיקת סטודנט ממערכת).

פקודה זו מקבלת פרמטר אחד: ID של סטודנט.  
מוחקת את התיקייה של הסטודנט וכל התוכן שלה. ומדפיסה הודעה מתאימה.

**Freeze** : (חסימת גישת סטודנט למערכת).

פקודה זו מקבלת פרמטר אחד: ID של סטודנט.  
נועלת את הגישה של הסטודנט על ידי הורדת הרשאות לתיקיה שלו.  
ומדפיסה הודעה מתאימה.

**Approve** : (ביטול חסימת סטודנט).

פקודה זו מקבלת פרמטר אחד: ID של סטודנט.  
משחררת נעילת הגישה של הסטודנט על ידי החזרת הרשאות לתיקיה שלו.  
ומדפיסה הודעה מתאימה.

**ShowCourses** : (מציגה קורסים זמינים)

פקודה זו לא מקבלת פרמטרים.  
מציגה רשימת קורסים זמנים לרישום. (תוכן הקובץ Courses.txt שנמצא בתיקיית Courses).

**Logout**

פקודה זו סוגרת את LecShell, וחוזרת ל-LockedShell.

## הנחיות חשובות:

- שימו לב, בכל פקודה שקשורה לתוכן של קבצים יש להשתמש רק בקריאות מ"ה לקבצים open,write,read.....
- בכל אחת מהפקודות יש להציג הודעת שגיאה אם הקריאה לא הופעלה כנדרש.
- בכל אחת מהפקודות שפותחת קובץ במקרה והקובץ לא קיים צריכים להדפיס הודעה מתאימה!
- לצורך נחות מניחים שכל הקבצים שנוצרו הם קבצי txt.
- על מנת ליצור תהליכים חדשים אתם חייבים להשתמש ב-fork ו-exec. אין להשתמש בקריאת system.
- אם התקבלה פקודה שלא נתמכת, עליכם להדפיס הודעה "Not Supported" למסך.
- כמו שהוזכר לעיל, כל פקודה חדשה תכתב כתכנית C נפרדה.
- אין להתייחס בתוכנית למסלולים אבסולוטיים לתיקיות שלכם בלבד. נדרש שהתוכנית שלכם תרוץ ותעבוד נכון ב /home/braude/.
- בקובץ הסקריפט להרצת shell תבצעו קימפול לכל תכניות ה C שנכתבו.

דוגמת הרצה (פלט התוכנית שלכם נדרש להיות כמו בדוגמה):

```
[braude@Cent ~]$ ./Sys_shell.sh
//Courses dir created, student dir created , all student dir created
in student dir.
LockShell>like
Not Supported
LockShell> LoginStudent Julia864 1111
Student Not Found!
LockShell> LoginStudent Julia8640 1111
Incorrect Password
LockShell> LoginStudent Julia8640 1111
Incorrect Password
LockShell> LoginStudent Julia8640 1111
Julia8640 account Blocked!
LockShell> LoginStudent Julia8640 11111
StdShell> ShowCourses
61589 deep_learning Monday 17:00 20:00
StdShell> MakeSchedule
Blocked Account ! permission denied
```

```
StdShell> Logout
LockShell> LoginStudent Sara2000 552255
StdShell> MakeSchedule
61589
61589 deep_learning Monday 17:00 20:00
Exit // Schedule.txt Not exist
StdShell> Logout
LockShell> LoginLecturer ChenK Aa123456
LecShell> GetSchedule 12345
Student Not Found.
LecShell> GetSchedule 112233445
Student Not register!
LecShell> SetCourses
SetCourses: missing parameters!
LecShell> SetCourses 61580 QA Sunday 10:00 13:00
Successfully created Course
// QA course Append to the end of the file Courses.txt
LecShell> Approve 123456789
Successfully Approved
LecShell> Logout
LockShell> LoginStudent Julia8640 11111
StdShell> GetNewCourses
61580 QA Sunday 10:00 13:00
StdShell> MakeSchedule
Insert Id Course:
61580
61580 QA Sunday 10:00 13:00
61589
61589 deep_learning Monday 17:00 20:00
Confirm // Save in Schedule.txt in 123456789 dir
Exit

StdShell> Logout
LockShell> exit
GoodBye...
```

דוגמאות לקבצים שנוצרו מפקודות שהורצו למעלה בדף הבא.

> MakeSchedule

שם קובץ: Schedule.txt				
Id	Name	day	start	end
61589	deep_learning	Monday	17:00	20:00

≥ SetCourses 61580 QA Sunday 10:00 13:00

שם קובץ: Courses.txt				
Id	Name	day	start	end
61589	deep_learning	Monday	17:00	20:00
61580	QA	Sunday	10:00	13:00

שם קובץ std_pass.txt			
Acount	Pass	ID	Name
Julia8640	11111	123456789	Julia
Sara2000	552255	112233445	Sara
Tiran164	555	554433221	Tiran

שם קובץ lec_pass.txt			
Acount	Pass	ID	Name
ChenK	Aa123456	123226789	Chen
ShellCol	6655433	554432211	Shelli
Ran1900	55555	554000221	Ran

- כל קובץ בעיצוב הנ"ל יתקבל.

**בהצלחה!**  
**צוות הקורס**