

# מערכות הפעלה – מטלה 1

## הוראות הגשה:

- יש להגיש את המטלה עד לתאריך 6.12.2018 בשעה 23:55.
- הגשת המטלה תהיה באמצעות תיבת ההגשה באתר המודל.
- עליכם להגיש קובץ zip ששמו יהיה **תעודת הזהות** שלכם ויכיל בתוכו שני קבצי C בשמות **ex11.c** ו-**ex12.c**. שימו לב! לא יתקבלו הגשות בקבצי rar ושמות שונים – היצמדו להוראות!
- יש להקפיד על coding style כפי שמקובל בשפת C.
- קראו בעיון את המטלה עד סופה טרם תחילת העבודה. שימו לב להערות ולהנחיות שבסוף התרגיל.
- עבודה ביחידים בלבד. ההעתיקה אסורה בהחלט.
- יש לוודא שהתכנית מתקמפלת ורצה ללא שגיאות/אזהרות. מומלץ מאוד להריץ בדיקות טרם הגשת התרגיל.

## חלק א' – השוואת תוכן קבצים

- שם התכנית: `ex11.c`

בחלק זה של התרגיל, עליכם לכתוב תכנית שתקבל את ה-`path` והשם של שני קבצים כפרמטרים ל-`main` ותבדוק האם הקבצים מכילים את אותו התוכן.

במידה ותוכן הקבצים זהה על התכנית להחזיר בסיום ריצתה (return value) את הערך 2. בכל מקרה אחר (התוכן של הקבצים אינם זהה או שקרתה שגיאה) על התכנית להחזיר את הערך 1. כמו כן, במידה והתכנית נכשלה יש להדפיס הודעת שגיאה מתאימה ל-`stderr`.

תוכלו לבדוק את נכונות התכנית שלכם על ידי כתיבת הפקודה `echo $?` בטרמינל לאחר הרצת התכנית. בהרצת פקודה זו תקבלו את הערך 2 במידה ותוכן הקבצים היה זהה, ואחרת תקבלו את הערך 1.

בעת קימפול התכנית תנו לה את השם `comp.out` במקום `a.out`. פעולה זו תוכלו לבצע ע"י הרצת הפקודה: `gcc ex11.c -o comp.out`

דוגמא להרצת התכנית:

```
./comp.out /home/dvir/workspace/files/f1.txt /home/dvir/workspace/files/f2.txt
```

## חלק ב' – בדיקת תכניות ומתן ציונים

### • שם התכנית: ex12.c

בחלק זה של התרגיל תכתבו תכנית המריצה תכניות אחרות, בודקת ונותנת להם ציון.

תכנית זו תקבל כפרמטר ל-main בשורת הפקודה שם ו-path של קובץ קונפיגורציה המכיל 3 שורות: שורה 1: מיקום של תיקייה המכילה תתי-תיקיות שבכל אחת מהן יכול להיות קובץ C (כל תת-תיקיה היא בעצם שם משתמש אחר).

שורה 2: שם ו-path של קובץ המכיל קלט.

שורה 3: שם ו-path של קובץ המכיל את הפלט הנכון עבור הקלט משורה 2.

על התכנית שלכם להכנס לכל תת-תיקיה שבתוך התיקיה משורה 1, לחפש בתקיה קובץ C, לקמפל את קובץ זה- C שמופיע בתקיה ולהריץ את קובץ ההרצה כתוצאה מהקימפול עם הקלט שמוכל במיקום משורה 2. על התכנית שלכם להשוות את הפלט של התכנית שהרצתם עם הפלט הרצוי שמיקומו מגיע משורה 3 של קובץ הקונפיגורציה בעזרת התכנית comp.out שממשתם בחלק א' (פשוט תריצו את התכנית comp.out כשהפרמטרים שלה הם הקובץ שנמצא בשורה 3 של קובץ הקונפיגורציה והפלט של התכנית של הסטודנט שאותה הרצתם).

התכנית שלכם צריכה לייצר קובץ CSV, בתקיה שממנה הורצה התכנית שלכם, ששמו יהיה results.csv. קובץ זה צריך להכיל עבור כל שם משתמש (שם של תת-תיקיה) את ציונו בהתאם לתשובה ש-comp.out החזירה (אפס או מאה) ואת הסיבה. יש לרשום את התו " " בין שם משתמש, לבין ציונו ולבין הסיבה.

סיבות אפשריות:

1. NO\_C\_FILE – אין בתקיה של הסטודנט קובץ עם סיומת c – ציון 0.
2. COMPILATION\_ERROR – בעיית קומפילציה – הקובץ אינו מתקמפל – ציון 0.
3. BAD\_OUTPUT – הפלט שונה מהפלט הרצוי – ציון 0.
4. GREAT\_JOB – הפלט זהה לפלט הרצוי – ציון 100.

דוגמא לקובץ results.csv:

```
Dvir,0,BAD_OUTPUT
Shir,0,COMPILATION_ERROR
Ron,100,GREAT_JOB
Liron,100,GREAT_JOB
Michael,0,NO_C_FILE
Hodia,0,BAD_OUTPUT
```

דוגמא לתוכן קובץ קונפיגורציה:

```
home/dvir/workspace/assignment1/students/
home/dvir/workspace/tests/input1.txt/
home/dvir/workspace/tests/output1.txt/
```

דוגמא לתוכן קובץ קלט משורה 2 של קובץ הקונפיגורציה:

```
6
3
```

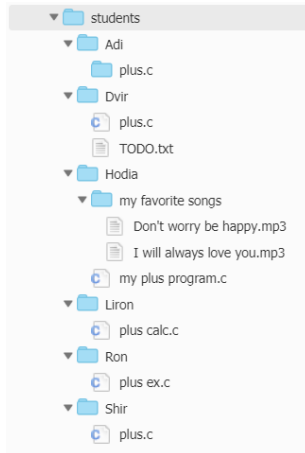
דוגמא לתוכן קובץ פלט רצוי משורה 3 של קובץ הקונפיגורציה:

```
6 plus 3 equals 9
```

דוגמא להפעלת קובץ הריצה של התכנית שלכם:

```
./a.out home/dvir/confFile.txt
```

דוגמא לתוכן של התיקייה מהשורה הראשונה של קובץ הקונפיגורציה:



## הערות והנחיות:

1. אתם רשאים לצאת מנקודת הנחה שב-path שהוזן בשורה הראשונה של קובץ הקונפיגורציה קיימות תקינות של סטודנטים בלבד, ולא קיים רעש אחר.
2. ניתן להניח שהתכניות שתריצו קולטות מה-stdin ופולטות ל-stdout ולכן תשתמשו ב-input/output redirection.
3. הניחו שבכל תת-תיקייה יש לכל היותר קובץ C אחד. יכול להיות שלא קיים קובץ C בכלל. שימו לב, היזהרו מלחשוב שתיקייה בשם "directory.c" זו תכנית בשפת C.
4. התכניות יכולות להדפיס למסך מה שהן רוצות, אין צורך להסתיר את ההדפסות.
5. הניחו שה-gcc מוגדר ב-path ולכן אין בעייה להשתמש ב-execvp.
6. השתמשו בקריאות המערכת fork, dup/dup2, exec, open, write, read, אסור להשתמש בפונקציות ספרייה אלטרנטיביות לפונקציות הנ"ל.
7. הניחו שקובץ ההרצה של התכנית comp.out נמצא בתקייה הנוכחית (./) שממנה יריץ הבודק את קובץ הריצה של חלק ב' של התרגיל (הבודק יקמפל תחילה את חלק א' של התרגיל שלכם ולאחר מכן את חלק ב').
8. יש לבדוק כל פונקצייה האם היא הצליחה או לא.
9. במידה וקריאת מערכת (SYSCALL) נכשלה יש להדפיס הודעת שגיאה ל-stderr ולסיים את התכנית בצורה נקייה.
10. כל שורה בקובץ הקונפיגורציה לא תעלה על 80 תווים.
11. אין חשיבות לסדר הסטודנטים שרשומים בקובץ הפלט results.csv.
12. חובה למחוק את כל הקבצים שהתכנית שלכם יצרה במהלך הריצה מלבד הקובץ results.csv. מחיקת הקבצים תעשה באמצעות קריאת המערכת unlink.
13. בכל תיקייה בלינוקס יש "." ו"-..", שימו לב לכך כאשר אתם משתמשים ב-readdir.

## בהצלחה!