ב"ה

תרגיל מס' 3 – תהליכים וקבצים

הוראות הגשה

- מועד אחרון להגשה: 25/05/20 23:59.
- יש לשלוח את הקבצים באמצעות האתר:
https://submit.cs.biu.ac.il/cgi-bin/welcome.cgi
לפני חלוף התאריך הנקוב לעיל.
- שם ההגשה של תרגיל 3 חלק א': ex31 ושל חלק ב': ex32
- **להזכירכם, העבודה היא אישית. "עבודה משותפת" דינה כהעתקה.**
- יש לוודא שהתרגיל מתקמפל ורץ על ה U2 ללא שגיאות/אזהרות.
- **שימו לב להערות בסוף התרגיל.**
- **בראשי שני קבצי ה-c אותם אתם מגישים, יש לכתוב שם מלא ות.ז. בפורמט הבא:**
//Israel Israeli 123456789

- **מענה לשאלות בנוגע לתרגיל יינתן בין התאריכים 11.5 עד 25.5 בלבד**, בפורום ייעודי שיפתח במודל.
- לפני פרסום שאלה יש לוודא שהשאלה לא נשאלה בעבר (ניתן להיעזר באופציית החיפוש במודל). שאלות חוזרות לא ייענו.

בהצלחה!

השוואת קבצים – חלק א'

<u>הנחיות עבור ex31</u>

- שם התרגיל: ex31
- שם קובץ מקור (source file) שיש לשלוח: ex31.c

כתבו תכנית המקבלת paths של שני קבצים כארגומנטים ל main ובודקת האם שני הקבצים זהים/דומים/שונים. למען הסר ספק – הנתיבים כוללים את שמות הקבצים.
אם הקבצים זהים (מכילים בדיוק אותו תוכן) התוכנית **תחזיר 1**, אם הקבצים דומים (הסבר בהמשך) התוכנית **תחזיר 3**, אחרת, אם הקבצים שונים- **תחזיר 2** (שימו לב, **לא מדובר כאן בתוכנית שמדפיסה למסך**).

קבצים זהים הם קבצים שכל התווים בהם זהים.
קבצים דומים הם קבצים שאינם זהים ובנוסף חופפים בלפחות חצי מהתווים שבהם. כלומר, אם נשים אותם אחד על השני בהיסטים כלשהם, תהיה חפיפה מוחלטת בלפחות חצי מהתווים.
במידה והקבצים אינם באותו גודל – דמיון יחושב לפי חפיפה בלפחות חצי מהתווים של הקובץ בעל המספר הנמוך יותר של תווים (הקובץ ה'קטן' יותר).
נבהיר, גם חפיפה בחצי מהתווים בדיוק תחשב שהקבצים דומים. על מנת ששני תווים יחשבו זהים הם צריכים להיות זהים במדויק, כלומר לא יתכן למשל שאחד מהם הינו אות קטנה והשני אות גדולה.
הקבצים עשויים להכיל אותיות (גדולות או קטנות), מספרים, רווחים, או ירידות שורה.
קבצים שונים הם קבצים שאינם זהים וגם אינם דומים.
דוגמא - עבור זוגות הקבצים הבאים – מצוין האם דומים או לא דומים.

<u>File A:</u>
ABCDEFGHI
<u>File B:</u>
CJREEFGHI
A and B are similar

<u>File C:</u>
ABCDEFGHI
<u>File D:</u>
CJREEFgHI
C and D are different

<u>File E:</u>
HELLO2WORLD
<u>File F:</u>
Hi3HELLO2WORLD
E and F are similar

File G:
A
cccCc

fF

File H:
fF
G and H are similar

File I:
Hello
File J:
Hello
I and J are identical

שימו לב, עבור חלק זה בתרגיל **חובה** להשתמש בפונקציה lseek.
לנוחיותכם, ניתן להשתמש ב-strcmp (למרות שבהערות שבסוף התרגיל צוין שאין להשתמש בפונקציות ספריה שניתן לממש באמצעות הכלים שלמדנו בתרגול).

כאשר אתם מקמפלים את התוכנית תנו לה את השם comp.out במקום a.out
דוגמא להרצת התוכנית:

```
[os2020@localhost ~]$ ./comp.out  /home/os2020/code/1.txt  /home/os2020/code/2.txt
```

לאחר הרצת התוכנית אם הקבצים זהים ונרשום את הפקודה ?$ echo נקבל את הערך 1, אם הם רק דומים נקבל את הערך 3, אחרת נקבל את הערך 2.

כלומר בהנחה שקבצים 1.txt ו 2.txt דומים:

```
[os2020@localhost ~]$./comp.out  /home/os2020/code/1.txt  /home/os2020/code/2.txt
[os2020@localhost ~]$ echo $?
3
[os2020@localhost ~]$
```

דוגמא נוספת:

```
[os2020@localhost ~]$./comp.out  /home/os2020/code/1.txt  /home/os2020/code/1.txt
[os2020@localhost ~]$ echo $?
1
[os2020@localhost ~]$
```

השוואת קבצים – חלק ב'

<u>הנחיות עבור ex32</u>

- שם התרגיל: ex32
- שם קובץ מקור (source file) שיש לשלוח: ex32.c

כתבו תכנית המקבלת path של קובץ קונפיגורציה כארגומנט ל main (בשורת הרצת התוכנית).
ניתן להניח שהקובץ קיים (ואינו תיקיה).
קובץ הקונפיגורציה הזה מכיל 3 שורות:
<u>שורה 1:</u> path של תיקייה המכילה תתי תיקיות (כל תת תיקייה זה בעצם שם משתמש) המכילות קבצי c. לא ניתן להניח שתיקיה זו תכיל רק תיקיות. בנוסף, לא ניתן להניח שהשורה הראשונה בהכרח תייצג תיקיה או שהיא מייצגת נתיב שאכן קיים במערכת הקבצים.
במידה ולא – יש להדפיס את ההודעה – Not a valid directory ולאחריה תו ירידת שורה.
לאחר מכן לצאת מהתוכנית ע"י החזרת 1-.
<u>שורה 2:</u> path של קובץ המכיל קלט.
<u>שורה 3:</u> path של קובץ המכיל את הפלט הנכון עבור קובץ הקלט משורה 2.

בהקשר לשורות 2 ו-3: לא ניתן להניח שהנתיבים אכן קיימים במערכת הקבצים. במידה והם לא קיימים, יש להדפיס Input/output File not exist ולאחר מכן תו ירידת שורה. לאחר מכן יש לצאת מהתוכנית ע"י החזרת 1-.
(ראו דוגמאות בהמשך).
קובץ הקונפיגורציה יסתיים בירידת שורה.

על התוכנית שלכם להיכנס לכל תתי התיקיות (ולהתעלם מקבצים אחרים שאינם תיקיות, במידה וקיימים) שבתוך התיקייה משורה 1, לחפש בתתי התיקיות שלה (ולא ברמות עמוקות יותר) קבצי c. לכל קובץ c שמופיע בתת-תיקיה – לקמפל אותו. את קובץ הריצה שנוצר יש להריץ עם הקלט שמופיע בקובץ שבמיקום שבשורה 2 (התוכנית שתריצו קולטת מ stdin ומדפיסה ל stdout ולכן תשתמשו ב i/o redirection).
הניחו שיהיה לכל היותר קובץ c אחד בתיקייה (יכול להיות שלא יהיה קובץ c בכלל). אין משמעות לשם של קובץ ה-c.
ניתן להניח שלקובץ ה-c תהיה בהכרח סיומת מתאימה, לדוגמא file.c

את הפלט של התוכנית יש להשוות עם הפלט הרצוי שמיקומו מגיע משורה 3 בעזרת התוכנית comp.out שממשתם בחלק א' של התרגיל (הריצו את התכנית comp.out ותתנו לה כארגומנטים ל main את המיקום של הפלט הנכון [שנמצא בקובץ הקונפיגורציה בשורה 3] ואת הפלט של התוכנית של המשתמש אותה הרצתם).

התוכנית שלכם צריכה לייצר קובץ (בתיקייה שממנה הורצה התוכנית שלכם) בשם results.csv שמכיל עבור כל שם משתמש (שם תת תיקייה) את ציונו בהתאם לתשובה ש comp.out החזירה (אפס עד מאה) ואת הסיבה. יש לרשום את התו "," בין שם המשתמש לבין הסיבה, לבין הציון שלו.

סיבות אפשריות:

1. NO_C_FILE - אין בתיקיה של המשתמש קובץ עם סיומת c. **הציון שינתן, 0.**
2. COMPILATION_ERROR –שגיאת קומפילציה (קובץ לא מתקמפל). **הציון שינתן, 10.**
3. TIMEOUT – קובץ ה c המקומפל רץ יותר מ 3 שניות. **הציון שינתן, 20.**

4. WRONG – הפלט שונה מהפלט הרצוי. **הציון שינתן, 50.**
5. SIMILAR – הפלט שונה מהפלט הרצוי אך דומה. **הציון שיינתן, 75.**
6. EXCELLENT – הפלט זהה לפלט הרצוי. **הציון שינתן, 100.**

שימו לב, כל סיבה עומדת בפני עצמה. כלומר, אין מצב שידרוש שרשור של סיבה אחת עם אחרת.

דוגמא לתוכן קובץ results.csv:

```
Chen,EXCELLENT,100
Dan,NO_C_FILE,0
Moshe,TIMEOUT,20
Dolev,COMPILATION_ERROR,10
Igor,WRONG,50
Guy,SIMILAR,75
```

דוגמא לתוכן קובץ הקונפיגורציה:

```
/home/os2020/students
/home/os2020/io/input.txt
/home/os2020/io/output.txt
```

דוגמא לתוכן קובץ קלט (שורה 2):

```
1
5 4
4
```

דוגמא לתוכן קובץ פלט (שורה 3):

```
Please enter an operation
Please enter two numbers
The sum is 9
Please enter an operation
Bye
```

דוגמא להפעלת קובץ הריצה של התוכנית שלכם:

```
[os2020@localhost ~]$ ./a.out /home/os2020/conf.txt
```

לנוחיותכם, מצ"ב לתרגיל זה תיקייה המכילה דוגמאות לבדיקה. תיקייה זו מכילה:

1. תיקייה students המכילה תיקיות של שמות משתמשים
2. קובץ קונפיגורציה- conf.txt
3. תיקיה בשם io ובה קובץ input וקובץ output
4. קובץ results.csv

הערות:

1. ב-ex31 אסור להדפיס למסך. ב-ex32 אפשר (כלומר אנחנו לא נבדוק הדפסות למסך בחלק זה של התרגיל).
2. **חובה לבדוק כל system call או פונקציית ספריה האם היא הצליחה או לא.**
אם היא לא הצליחה יש לכתוב הודעה אינפורמטיבית מתאימה (שתנוסח לבחירתכם, אלא אם הוגדר אחרת בתרגיל) בעזרת הפונקציה write ל file descriptor מספר 2 (stderr).
נחריג את הפונקציה write – גם עבורה יש לבדוק כישלון, אך אם היא נכשלה אין להדפיס הודעת שגיאה או לנסות לקרוא לה שוב.
בנוסף להדפסת השגיאה:
   - ב-ex31 יש לצאת מהתוכנית לאחר מכן (בצורה נקייה כמובן) ולהחזיר 1-.
   - ב-ex32 אם השגיאה קרתה בשלב כלשהו לאחר כניסה מוצלחת לתת תיקייה של התיקייה הראשית שצויינה בקובץ הקונפיגורציה- יש להמשיך בתוכנית לטיפול בתת התיקייה הבאה (אם קיימת).
   אחרת- לצאת מהתוכנית לאחר מכן (בצורה נקייה כמובן) ולהחזיר 1-.
3. הניחו שה gcc מוגדר במשתנה הסביבה PATH ולכן אין בעיה להשתמש ב execvp.
4. אתם רשאים להשתמש בכל קריאות המערכת שנלמדו בתרגולים עד היום. במידה וניתן לבצע פעולה באמצעות קריאות המערכת שלמדנו, אסור להשתמש בפונקציות אלטרנטיביות אחרות, כמו לדוגמא fread במקום read. **אין להשתמש בפונקציה sleep.**
5. הניחו שהתכנית comp.out תהיה בתקיה הנוכחית (./) שממנה יריץ הבודק את קובץ הריצה של חלק ב' של התרגיל (הבודק יקמפל את החלק הראשון ורק אח"כ את החלק השני).
6. כל שורה בקובץ הקונפיגורציה לא תעלה על 150 תווים. אתם יכולים גם להניח שאורך ה path המקסימלי המוביל לקובץ C גם לא יעלה על 150 תווים.
7. אין חשיבות לסדר של הסטודנטים הרשומים בקובץ results.csv
8. חובה למחוק את כל הקבצים שהתוכנית שלכם יצרה (חוץ מהקובץ results.csv) בסוף הריצה.
9. אם תכנית של סטודנט לא מסיימת את עבודתה תוך **3 שניות** אתם לא צריכים לדאוג לסיים אותה, אך אתם צריכים להתייחס לזה בקובץ results.csv ע"י ציון 20 וסיבה תואמת.
10. בכל תיקיה בלינוקס יש "." ו ".." (קישור לתיקיה הנוכחית ולתיקיית האב), שימו לב לכך כאשר אתם משתמשים ב readdir.
11. יש לסגור file descriptors שנפתחו על ידכם.
12. שימוש במשתנים גלובאליים הינו אסור.
13. אין צורך להשתמש בהקצאות דינאמיות בתרגיל.

בהצלחה!