

מבוא למדעי המחשב – שפת פייתון (234128)

חורף 2024-2025

תרגיל בית 4

מועד הגשה: 20/01/2025 בשעה 20:00

הגשה רק בבודדים

חשוב לשמור את קוד
אישור ההגשה שמקבלים
מהמערכת כשמגישים!!

המתרגל האחראי על תרגיל זה: סהר אדמוני

E-mail: saharad@campus.technion.ac.il

שעת קבלה רגילה: יום ב 15:30-16:30 ,

שעות קבלה מיוחדות לשאלות על התרגיל:

יום ב' (13.01.2025) 16:30-17:30 , <https://technion.zoom.us/j/99419545690>

וביום ג' (16.01.2025) 10:30-11:30 , במשרד דייוויס 105

מילואים:

- **מילואים:** שיטת ההקלות של קורס פייתון מותאמת לסילבוס של הקורס. **אין להסתמך בשיטות בקורסים אחרים!**
- לטיפול בבקשות הארכה להגשה או ביטול הגשה או עזרה בחומר הקורס יש לפנות ל**ילנה** (enovbari@gmail.com), המתרגלת האחראית של הקורס.
- באופן רגיל למילואים לא ממושכים כל יום מילואים = יום דחייה. על מנת לקבל דחייה עבור ת"ב 4, יש לשלוח מייל בצירוף אישור מילואים (או מסמך שקול) למתרגלת אחראית ילנה.
- בונוס של 5 נקודות ניתן לסטודנטים/יות שמגישים אלקטרונית דרך אתר הקורס, בזמן עד למועד הגשה. לא ניתן טיפול ידני בבונוס (ללא קשר לנסיבות ואישורים).

הוראות הגשה:

- הגשה אלקטרונית **בבודדים**.
- תרגיל שיוגש בזמן (עד למועד ההגשה ולא בתקופת הארכה) יקבל אוטומטית **בונוס של 5 נקודות**.
- תאריך הגשת ערעורים יתפרסם עבור כל תרגיל הבית לחוד.
- **ערעורים על הדפסות שגויות בפלט שלכם לא יתקבלו**. עליך לבדוק את תוכנית שלך עם DiffMerge לפני ההגשה.

הנחיות:

- אנא קרא/י את השאלות בעיון לפני שתתחיל/י בפתרון.
- ניתן להניח כי הקלט תקין, אלא אם כן נאמר אחרת.

- **שימו לב! הבדיקה תהיה אוטומטית, ולכן הקפידו/י מאוד להדפיס בדיוק בפורמט שהתבקשת ולבדוק את הפלט מול הפלט של הדוגמאות שקיבלת עם DiffMerge.**

○ וודאו/י את האותיות הגדולות והקטנות לפי הדוגמאות וההסברים בתרגיל.

- יש לבדוק את אתר ה-FAQ של התרגיל לעיתים תכופות על מנת לקבל עדכונים על התרגיל.
- הגשה אלקטרונית ב**בודדים** היא דרך אתר הקורס. הגיש קובץ **hw4.zip** (zip) **ולא** פורמט אחר (!) כגון rar). ה**שם** של קובץ חייב להיות **באנגלית ללא רווחים**. אין דרישות נוספות לשם הקובץ, כלומר גם sharon.zip מקובל מבחינת המערכת. הקובץ יכיל בתוכו **רק** את הקבצים הבאים (לא לעשות zip על התיקיה בה נמצאים הקבצים להגשה – במקרה כזה ההגשה לא תתקבל ע"י המערכת!):

○ קובץ **students.txt** עם שמך באנגלית, מספר תעודת הזהות וכתובת האימייל שלך.

○ קובץ פתרון **hw4q1.py** עבור שאלה 1.

○ קובץ פתרון **hw4q2.py** עבור שאלה 2.

- השתמש ב-redirection כדי להפנות את הפלט לקובץ טקסט.
- יש להקפיד להגיש את כל הקבצים בדיוק עם השמות שמופיעים לעיל. הגשה שלא תעמוד בתנאי זה לא תתקבל ע"י המערכת!

שימו לב לאתר ה-FAQ של התרגיל, אשר מכיל עדכונים ותשובות לשאלות נפוצות, ויעודכן מפעם לפעם. חשוב להתעדכן לעיתים תכופות – זה יכול לחסוך הרבה בעיות, ונקודות שירדו בבדיקה שכשלה. בפרט, לפני פנייה למתרגל האחראי בדקו את הכתוב על מנת לחסוך לעצמכם זמן המתנה לתשובה, שבדרך כלל תהיה הפניה ל-FAQ בכל מקרה. ניתן למצוא את ה-FAQ בכתובת:

<https://docs.google.com/document/d/1sSMelrYmeW4jdCRut0aGamh96F1EuVNIEjOOOf8bqTBu/edit?usp=sharing>

חובה להשוות את פלט התכנית שלכם לפי קבצי הדוגמאות שמסופקים לכם בעזרת התוכנה DIFFMERGE

! חובה להתעדכן בקובץ השאלות הנפוצות !

<https://docs.google.com/document/d/1sSMelrYmeW4jdCRut0aGamh96F1EuVNIEjOOOf8bqTBu/edit?tab=t.0>

ת"ב 4 מבוסס על תרגולים 7,8. אין להשתמש בחומר שנלמד אחריהם.

שאלה 1

בסוף השאלה מופיעות החתימות של כל הפונקציות – תקראו את כל השאלה לפני שאתם מתחילים לכתוב תשובות.
שימו לב - בשאלה זו 3 סעיפים.

סעיף א:

עליכם לכתוב פונקציה המקבלת מחרוזת (my_str), תו (char) ומספר (num) ומחזירה (return) מחרוזת חדשה שזהה למחרוזת המקורית (my_str) פרט לכך שאם char אות קטנה אז num ההופעות הראשונות של התו char הופכות לאותיות גדולות (אם קיימות).

הערה 1: אם num=0 אז מוחזרת המחרוזת המקורית

הערה 2: אם התו לא מופיע במחרוזת אז מוחזרת המחרוזת המקורית

הערה 3: אם num גדול ממספר ההופעות של התו במחרוזת אז כל ההופעות של התו במחרוזת הופכות לאותיות גדולות (ראו דוגמה 3)

הערה 4: אם מופיע במחרוזת תו מיוחד (תו שאינו אות קטנה) והתו char הוא זהה לאותו תו מיוחד, התו יישאר כפי שהוא במחרוזת ללא שינוי (ראו דוגמה 4)

דוגמה 1:

קלט:

"abcc", "c", 1

פלט:

"abCc"

דוגמה 2:

קלט:

"abcc", "c", 2

פלט:

"abCC"

דוגמה 3:

קלט:

"abcc", "c", 3

פלט:

"abCC"

דוגמה 4:

קלט:

"abc\$", "\$", 1

פלט:

"abc\$"

דוגמה 5:

קלט:

"aBbb", "b", 1

פלט:

"aBBb"

סעיף ב:

כתבו פונקציה המקבלת רשימה של מחרוזות (lst), תו (char) ומספר (num). הפונקציה מעדכנת את איברי הרשימה כך שבכל מחרוזת ברשימה num ההופעות הראשונות של התו char הופכות לאותיות גדולות.

הערה 1: אתם יכולים להניח שברשימה יש רק מחרוזות

הערה 2: הפונקציה לא מחזירה או מדפיסה שום דבר. רק מעדכנת את הרשימה שהתקבלה

הערה 3: חובה להשתמש בפונקציה מסעיף א'

הערה 4: אין להדפיס את הרשימה המעודכנת

דוגמה 1:

קלט:

["abcc", "abc", "cagfc"], "c", 1

הרשימה המעודכנת:

["abCc", "abC", "Cagfc"]

דוגמה 2:

קלט:

["abcc", "abc", "cagfc"], "f", 1

הרשימה המעודכנת:

["abcc","abc","cagFc"]

דוגמה 3:

קלט:

["abcc","abc","cagfc"], "c",4

הרשימה המעודכנת:

["abCC","abC","CagfC"]

סעיף ג:

בסעיף זה עליכם לכתוב פונקציה שמקבלת מהמשתמש לפי הסדר הבא:

1. מחרוזת עם פסיקים str_list

2. תו char

3. מספר num

הפונקציה ממירה את המחרוזת עם הפסיקים לרשימה שאיבריה הם תת-מחרוזות המופרדות ע"י פסיק. עליכם להשתמש בסעיפים הקודמים כדי לעדכן את הרשימה כך שבכל מחרוזת num ההופעות הראשונות של התו char יהפכו לאותיות גדולות.

לאחר עדכון הרשימה הפונקציה **מדפיסה** את הרשימה המעודכנת.

הערה 1: אתם יכולים להניח שהמשתמש לא ייתן קלטים לא תקינים.

הערה 2: המחרוזת עם הפסיקים יכולה להיות ריקה ("") – במקרה כזה מודפסת רשימה ריקה

הערה 3: התו יכול להיות מחרוזת ריקה ("") – במקרה כזה מוחזרת הרשימה המתאימה

למחרוזת ללא שינוי

דוגמה 1:

קלט:

"abcc,abc,cagfc"

"c"

1

הדפסה:

["abCc","abC","Cagfc"]

דוגמה 2:

קלט:

"abcc,abc,cagfc"

"f"

1

הדפסה:

["abcc","abc","cagFc"]

דוגמה 3:

קלט:

"abcc,abc,cagfc"

"c"

4

הדפסה:

["abCC","abC","CagFc"]

לפני ההגשה מחברת jupyter notebook שלכם צריכה להראות כך:

```
def changeToUpper(my_str, char, num):  
    # code here
```

```
def changeToUpperFromList(lst, char, num):  
    # code here
```

```
def main():  
    # code here
```

```
main()
```

המלצה:

כדי לבדוק את הפונקציות בסעיף א' וב' אפשר לקרוא להן ולהעביר מידע מתאים.

חשוב! לפני ההגשה תמחקו את כל הבדיקות הללו - הן רק בשבילכם. תגישו את הקוד לפי החתימות בתמונה למעלה.

לדוגמה, עבור סעיף א:

```
my_str="abcc"  
char="c"  
num=1  
print(changeToUpper (my_str, char, num))
```

לדוגמה, עבור סעיף ב:

```
lst = ["abcc", "abc", "cdsc"]  
char="c"  
num=1  
changeToUpperFromList (lst, char, num)  
print(lst)
```

שאלה 2

בסוף השאלה מופיעות החתימות של כל הפונקציות – תקראו את כל השאלה לפני שאתם מתחילים לכתוב תשובות. לאורך כל השאלה אתם יכולים להניח שכל הרשימות (מטריצות ולא מטריצות) המתקבלות הן של מספרים שלמים (int) ושכל הקלטים תקינים. שימו לב - בשאלה זו 3 סעיפים.

סעיף א:

כתבו פונקציה המקבלת שתי רשימות מספרים באותו האורך - הראשונה רגילה והשנייה תייצג פרמוטציה. על הפונקציה **לשנות** את הרשימה הרגילה כך שכל איבר יעבור למיקום החדש שלו לפי רשימת הפרמוטציה.

הערה 1: שתי הרשימות בהכרח באותו האורך ולא ריקות

הערה 2: הערכים ברשימת הפרמוטציה בהכרח תקינים. כלומר, אלו אינדקסים שבהכרח מתאימים לרשימה הרגילה

הערה 3: הפונקציה לא מחזירה או מדפיסה שום דבר. רק מעדכנת את הרשימה שהתקבלה

דוגמה 1:

קלט:

[1,2,3] [1,2,0]

הרשימה המעודכנת:

[3,1,2]

הסבר: רשימת הפרמוטציה היא [1,2,0] - המשמעות היא:

- האיבר במיקום 0 (1) צריך לעבור למיקום 1
- האיבר במיקום 1 (2) צריך לעבור למיקום 2
- האיבר במיקום 2 (3) צריך לעבור למיקום 0

התוצאה היא [3,1,2] כי:

- 1 עבר למיקום 1
- 2 עבר למיקום 2
- 3 עבר למיקום 0

דוגמה 2:

קלט:

[1,2,3,4,5,6] [1,2,3,0,4,5]

[4,1,2,3,5,6]

סעיף ב:

כתבו פונקציה המקבלת מהמשתמש מטריצה באופן הבא: תחילה, היא תקבל מספר שייצג את **מספר השורות במטריצה**. לאחר מכן, היא תקבל את השורות המטריצה בזו אחר זו באופן הבא - כל שורה במטריצה תתקבל על ידי רצף מספרים המופרדים ע"י פסיקים ובין שורה לשורה יהיה enter (ירידת שורה). הפונקציה **תחזיר** (return) את המטריצה המתקבלת – **מטריצה של מספרים שלמים (int) ולא מחרוזות של מספרים**.

הערה 1: לא ייתכן מספר שורות 0

הערה 2: הקלט תקין- מספר השורות שיוזנו בהכרח יתאים למספר המייצג את מספר השורות במטריצה.

דוגמה 1:

קלט:

3

1,2,3

4,5,6

7,8,9

פלט:

[[1, 2, 3], [4, 5, 6], [7, 8, 9]]

דוגמה 2:

קלט:

3

2,5,6,7

4,7,8,9

10,12,100,18

פלט:

[[2, 5, 6, 7], [4, 7, 8, 9], [10, 12, 100, 18]]

סעיף ג

עליכם לכתוב פונקציה המבצעת פרמוטציה על שורות המטריצה ומדפיסה אותה באופן הבא:

1. לקבל מהמשתמש מטריצה – תחילה יתקבל מספר השורות ואז השורות בזו אחר זו
 2. לקבל מהמשתמש רשימת פרמוטציה
 3. לבצע פרמוטציה על שורות המטריצה לפי רשימת הפרמוטציה, כאשר: כל מספר i ברשימת הפרמוטציה מציין שהשורה במיקום i במטריצה המקורית צריכה לעבור למיקום שמצוין בערך שנמצא במיקום i ברשימת הפרמוטציה
 4. **להדפיס** את המטריצה לאחר שעברה פרמוטציה
- הערה 1: חובה להשתמש בסעיפים הקודמים- שימו לב לסדר הקלטים המתקבלים מהמשתמש
- הערה 2: לא לשכוח להמיר את האיברים בקלטים ל - `int`
- הערה 3: כל הקלטים תקינים – לא ייתכן מספר שורות 0 ולא ייתכן רשימות ריקות

דוגמה 1:

קלט:

3

1,2,3

4,5,6

7,8,9

1,2,0

הקלט – (1 מספר שורות; 2 שורות של המטריצה לפי המספר; 3) רשימת פרמוטציה

הדפסה:

7,8,9

1,2,3

4,5,6

הסבר: רשימת הפרמוטציה היא $[1,2,0]$ - המשמעות היא:

- שורה במיקום 0 (שורה ראשונה: $[1,2,3]$) צריכה לעבור למיקום 1
- שורה במיקום 1 (שורה שנייה: $[4,5,6]$) צריכה לעבור למיקום 2
- שורה במיקום 2 (שורה שלישית: $[7,8,9]$) צריכה לעבור למיקום 0

דוגמה 2:

קלט:

2
1,2,3,4
4,5,6,7
1,0

הדפסה:

4,5,6,7
1,2,3,4

לפני ההגשה מחברת jupyter notebook שלכם צריכה להראות כך:

```
def applyPermutation(num_lst, perm_lst):  
    # code here
```

```
def getMatrix():  
    # code here
```

```
def permuteMatrix():  
    # code here
```

```
permuteMatrix()
```

המלצה:

כדי לבדוק את הפונקציות בסעיף א' וב' אפשר לקרוא להן ולהעביר מידע מתאים.
חשוב! לפני ההגשה תמחקו את כל הבדיקות הללו- הן רק בשבילכם. תגישו את הקוד לפי החתימות בתמונה למעלה.

לדוגמה עבור סעיף א:

```
num_lst=[1,2,3]  
perm_lst=[1,2,0]  
applyPermutation(num_lst,perm_lst)  
print(num_lst)
```

לדוגמה עבור סעיף ב:

```
print(getMatrix())
```

בהצלחה!!!