

מבוא למדעי המחשב – שפט פיתון (234128)

chorf 2024-2025

תרגיל בית 4

מועד הגשה: 20/01/2025 בשעה 20:00

הגשה רק בבודדים

חשוב לשמור את קוד
אישור הגשה שמקבלים
המערכת שימושים!!

המתרגל האחראי על תרגיל זה: סהר אדמוני

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

שעת קבלה רגילה: יומם ב' 15:30-16:30,

שעות קבלה מיוחדות לשאלות על התרגיל:

יום ב' (13.01.2025) 16:30-17:30

וביום ג' (16.01.2025) 10:30-11:30, במשדר דיוויס 105

AMILAIM:

- AMILAIM: שיטת ההקלות של קורס פיתון מותאמת לSilver של הקורס. אין להסתמך בשיטות בקורסים אחרים!

לטיפול בבקשות הארכה להגשה או ביטול הגשה או עזרה בחומר הקורס יש לפנות [לונה](#) (enovbari@gmail.com), המתרגלת האחראית של הקורס.

- באופן רגיל לAMILAIM לא ממושכים כל יוםAMILAIM = יומם דחיה. על מנת לקבל דחיה עברו ת"ב 4, יש לשלוח מייל ביצירוף אישורAMILAIM (או מסמך שקוול) למתרגלת האחראית לינה.
- בונוס של 5 נקודות ניתן לסטודנטים/יות שmagisim אלקטטרונית דרך אתר הקורס, בזמן עד למועד הגשה. לא ניתן טיפול יدني בבונוס (לא קשר לנסיבות או אישורים).

הוראות הגשה:

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

הנחיות:

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

שים לב! הבדיקה תהיה אוטומטית, ולכן הקפיד מאוד להדפיס בפורמט שהתקבש
ולבדוק את הפלט מול הפלט של הדוגמאות שקיבלת עם DiffMerge.

- וודאי את האותיות הגדולות והקטנות לפני הדוגמאות וההסברים בתרגיל.
- יש לבדוק את אתר ה-FAQ של התרגיל לעיתים תכופות על מנת לקבל עדכונים על התרגיל. הגשה אלקטרוני**ם** בבודדים היא דרך הקורס. הגיש קובץ **hw4.zip** (zip ולא פורמט אחר (!) כגון rar). השם של קובץ חייב להיות **אנגלי ללא רווחים**. אין דרישות נוספת לשם הקובץ, כלומר גם zip.sharon.mkobbel מביחסת המערכת. הקובץ יכול בתוכו **הן** את הקבצים הבאים (לא לעשות zip על התקייה בה נמצאים הקבצים להגשה – במקרה זה הגשה לא תתקבל ע"י המערכת!):
- קובץ **students.txt** עם שמר אנגלית, מספר תעודה זהה וכתובת האימייל שלו.
- 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 בדיק עם השמות שמופיעים לעיל. הגשה שלא תעמוד בתנאי זה לא תתקבל ע"י המערכת!

שים לב לאתר ה-FAQ של התרגיל, אשר מכיל עדכנים ותשובות לשאלות נפוצות, ויעודכן מפעם לפעם. חשוב להתעדכן לעיתים תכופות – זה יכול להשוו הרבה בעיות, ונΚודות שירדו בבדיקה שכלה. בפרט, לפני פניה למתרגל האחראי בדק 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/1sSMelrYmeW4jdCRut0aGamh96F1EuVNIEjOOOf8bqTB
U/edit?usp=sharing](https://docs.google.com/document/d/1sSMelrYmeW4jdCRut0aGamh96F1EuVNIEjOOOf8bqTBU/edit?usp=sharing)

חוּבָה לְהַשׁוֹת אֶת פְּלֵט הַתְּכִנִּית שְׁלַכְתָּ לְפִי קְבָצֵי הַדּוֹגְמָוֹת שְׁמוֹפְּקִים לְכֶם בְּעִזּוֹת הַתוֹּכָנָה DIFFMERGE

חוּבָה לְהַתְּעַדְּכָן בְּקֻבָּץ הַשְּׁאֲלוֹת הַנְּפֹוצֹת !

<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 מתחלים לכתוב תשובה.
שים לב - שאלה זו 3 סעיפים.

סעיף א:

עליכם לכתוב פונקציה המתקבלת מחזרת (my_str),תו (char) ומספר (num) **ומחזירה** (return) מהזרת חדשה שזאה למחזרת המקורי (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 המחרצת המקורי

הערה 2 : אם התו לא מופיע במחזרת אז מוחזרת המחרצת המקורי

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

הערה 4 : אם מופיע במחזרתתו מיוחד (תו שאינו אות קטנה) והטו char זהה לאותו לתו מיוחד, התו יישאר כפי שהוא במחזרת 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 שכל מחרוזת ברשימה num הופעתה 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 עליום לכתוב פונקציה שמקבלת מהמשתמש לפי הסדר הבא:

1. מחזורות עם פסיקים str_list

2. תו char

3. מספר num

הfonkzia מmirah את המחרוזות עם הפסיקים לשישמה שאיבריה הם תת-מחזרות המופרדות ע"י פסיק. עליום להשתמש בסעיפים הקודמים כדי לעדכן את הרשימה כך שבכל מחרוזת num ההופעות הראשונות של התו char ייפכו לאותיות גדולות.

לאחר עדכן הרשימה הפונקציה **מdfsah** את הרשימה המעודכנת.

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

הערה 2: המחרוזות 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

הdfsah:

[“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()
```

הملצתה:

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

חשוב! לפני ההגשה תמחקו את כל הבדיקות הללו - הן רק בשביבכם. תגיסו את הקוד לפי החתימות בתמונה לעיל.

לדוגמה, עברו 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 – תקרוו את כל השאלה לפני שאתם מתחילה לכתוב תשובה. לאורך כל השאלה אתם יכולים להניח שככל הרשימות (מטריצות ולא מטריצות) המתקבלות הן של מספרים שלמים (int) וscal הקלטים תקין.

שים לב - בשאלת זו 3 סעיפים.

סעיף א:

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

הערה 1: שתי הרשימות בהכרח באותו האורך ולא ריקות

הערה 2: הערכים ברשימה הפרמטרציה בהכרח תקין. למשל, אלו אינדקסים שבהכרח מתאימים לרשימה הרגילה

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

דוגמה 1:

קלט:

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

הרשימה המעודכנת:

[3,1,2]

הסבר: רישימת הפרמטרציה 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]

הרשימה המעודכנת:

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

סעיף ב:

כתבו פונקציה המקבלת מהמשתמש מטריצה באופן הבא: תחילה, היא מקבל מספר שמייצג את **מספר השורות במטריצה**. לאחר מכן, היא מקבל 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 המתתקבלת – **מטריצה של מספרים שלמים (int) ולא מחוזות של מספרים**.

הערה 1: לא יתכן מספר שורות 0

הערה 2: הקלט תקין- מספר השורות שיוזנו בהכרח יתאים למספר המציין 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 בזה אחר זו
 2. לקבל מהמשתמש רשימת פרמוטציה
 3. לבצע פרמוטציה על שורות המטריצה לפי רשימת הפרמוטציה, כאשר: כל מספר ? ברשימת הפרמוטציה מצין שהשורה במיקום ? במטריצה המקורית צריכה לעבור למיקום שמצוין בערך שנמצא במיקום ? ברשימת הפרמוטציה
 4. להדפיס את המטריצה לאחר שעבירה פרמוטציה
- הערה 1: חובה להשתמש בסעיפים הקודמים - שימוש לב לסדר הקלטים המתקבלים מהמשתמש
הערה 2: לא לשכוח להמיר את האיברים בקלטים ל - int
הערה 3: כל הקלטים תקין – לא יתכן מספר שורות 0 ולא יתכן 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 שלישיית: [7,8,9]) צריכה לעבור למיקום 0

דוגמה 2 :

קליט:

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

הדף:

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

לפניהם הגשה מחברת jupyter notebook שלכם צריכה להראות כך:

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

def getMatrix():
    # code here

def permuteMatrix():
    # code here

permuteMatrix()
```

המלצתה:

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

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())
```

בצלחה!!!