

מבוא

בתרגיל זה נתרגל שימוש בלולאות ומשתנים. דגשים לתרגיל:

בתרגיל זה אין להשתמש באינדקסים שליליים ו/או באינדקסים חלקיים בגישה לרשימות ו/או מחרוזות:
כלומר, פקודות מהצורה מותרות

`my_list[start:step]`
`[my_start[idx`
`my_start[start:stop:step]`

אבל כל הצורות הבאות (או דומות להם) אסורות:

`my_list[start:]`
`my_list[:end]`
`my_list[::step]`
`my_list[-idx]`
`my_list[:,idx]`

- כתבו את כל הקוד שלכם בתוך הקובץ `ex3.py`.
- **חתימות הפונקציות** (שם הפונקציה והפרמטרים שלה) במימוש שלכם צריכות להיות זהות במדויק לחתימות המתוארת במשימות. ניתן לשנות את חתימות הפונקציות על ידי הוספת ערכים דיפולטיביים לפרמטרים של פונקציות (במקומות בהם השימוש מתאים).
- לנוחותכם, מסופק **קובץ שלד** הכולל את כל חתימות הפונקציות שיש לממש.
- ניתן להוסיף לקובץ המוגש **פונקציות נוספות** במידת הצורך וניתן להשתמש בשאלות מתקדמות בפונקציות שמימשותם בסעיפים קודמים.
- **סגנון:** הקפידו על תיעוד נאות ובחרו שמות משתנים משמעותיים. הקפידו להשתמש בקבועים (שמות משתנים באותיות גדולות) על פי הצורך.
- שימוש בכלים **שלא נלמדו בקורס** – ניתן להשתמש בכלים אשר לא נלמדו בקורס כל עוד הם לא מייתרים את השאלה. לדוגמה, אם המשימה באחת השאלות הייתה לממש סכום של איברים ברשימה באמצעות לולאות, ניתן לייתר את השאלה על ידי שימוש בפונקציה `sum` המובנית של Python כדי להשיג את התוצאה הנ"ל. במקרה של ספק, יש לשאול בפורום בצורה קונקרטית. בפתרון התרגיל **אין להשתמש בפונקציות** – `count`, `index`, `sum`, במודולים של `numpy` וב – `itertools`.
- **רשימה ריקה** – היא רשימה אשר אינה מכילה איברים []. היא עדיין נחשבת רשימה ומספר האיברים בה הוא 0.
- **מחרוזת ריקה** – היא מחרוזת אשר אינה מכילה תווים: "" ו - ". היא עדיין נחשב מחרוזת ואורכה הוא 0. מחרוזת המכילה רוח אחד " " או מספר רווחים " " היא אינה מחרוזת ריקה.
- שימו לב מתי יש לקבל **קלט מהמשתמש** בעזרת הפונקציה `input` (השאלה הראשונה בלבד) ומתי הקלט מתקבל כארגומנט בעת קריאה לפונקציה. כאשר הפונקציה לא מקבלת קלט מהמשתמש בעזרת פונקציית `input`, מומלץ לכתוב בעצמכם קטעי קוד אשר קוראים לפונקציות עם ארגומנטים מתאימים כדי לבדוק שהן עובדות (אין לכלול את קטעי קוד הבדיקה בהגשה הסופית).

- בכל שאלה מפורט מה ניתן להניח על הקלט - אין צורך לבצע בדיקות תקינות נוספות מעבר למפורט (כלומר, ניתן להניח כי הקלט תקין).
- בכל הסעיפים, כאשר פונקציה צריכה להחזיר ערך, הכוונה היא לשימוש במילה השמורה `return`, בפרט הכוונה אינה להדפיס למסך בעזרת הפונקציה `print`. שימו לב, אין להדפיס למסך כל הודעה מלבד ההודעות הנדרשות במפורש.
- בפונקציות המקבלות רשימות כקלט – אין לשנות את רשימות הקלט.

חלק א': קלט מהמשתמש

1. רוני החליטה לשפר את המימוש של פונקציות `fancy_arithmetic_mean` (שממשותם בתרגיל הקודם) ולכתוב פונקציה חדשה המחשבת את כל סוגי הממוצעים עבור מספר לא מוגבל של מספרים. כתבו פונקציה המממשת את האפליקציה המשופרת לפי ההנחיות הבאות:

- הפונקציה תיקרא `fancy_mean`
- הפונקציה לא מקבלת פרמטרים.
- לאחר קריאה לפונקציה, הפונקציה תדפיס למסך את ההודעה (שימו לב, לאחר הנקודתיים בסוף השורה רווח יחיד):

Please enter the numbers, one in each line:

ותעבור לשורה חדשה.

- הפונקציה תמתין לקלט של המספרים (אחד בכל שורה, לפי הבחירה לעיל).
- כאשר ילחץ `Enter` ללא מספר, הפונקציה לא תחכה למספרים נוספים.
- אם לא הוכנס אף מספר, הפונקציה תחזיר `None`.
- אחרת תודפס הודעה מהצורה הבאה (לדוגמה, אם הוכנסו המספרים 4 ו-9):

The arithmetic mean of the numbers is 6.5

The geometric mean of the numbers is 6

The harmonic mean of the numbers is 5.538461538461538

- אפשר להניח קלטים תקינים עבור הממוצע הגאומטרי וההרמוני.
- ניתן להשתמש בפונקציות שכתבתם בתרגיל הקודם אך אין הכרח לעשות זאת.
- תזכורת: בהנתן n מספרים (x_1, \dots, x_n) , הממוצע הגאומטרי וההרמוני נתונים ע"י הנוסחאות

$$\text{Geometric mean} = \sqrt[n]{x_1 \cdot \dots \cdot x_n}, \quad \text{Harmonic mean} = \frac{n}{\frac{1}{x_1} + \dots + \frac{1}{x_n}}$$

דוגמאות להרצה:

Please enter the numbers, one in each line:

1
2
3

The arithmetic mean of the numbers is 2.0

The geometric mean of the numbers is 1.8171205928321397

The harmonic mean of the numbers is 1.6363636363636365

0.5,1,4,3:

The arithmetic mean of the numbers is 2.125

The geometric mean of the numbers is 1.5650845800732873

The harmonic mean of the numbers is 1.1162790697674418

חלק ב': לולאות ורשימות

2. פלינדרום הוא רצף או רשימה אשר נראים אותו הדבר אם עוברים על האיברים אחד אחד מההתחלה או מהסוף. לדוגמה, המחרוזת "פרשנו רעבתן שבדבש נתבער ונשרף" היא פלינדרום, כמו גם הרשימה הריקה, רשימה עם איבר אחד, או הרשימה [2, 3, 4, 3, 2]. כתבו פונקציה המקבלת רשימה ומחזירה True אם הרשימה היא פלינדרום, אחרת הפונקציה מחזירה False, על פי ההנחיות הבאות:
- במימוש הפתרון אין לשנות את הרשימה הניתנת כקלט.
 - חתימת הפונקציה:

`def is_palindrome(lst):`

- הנחות על הקלט:
 - o `lst` היא רשימה. לא ניתן להניח כי הרשימה מכילה איברים.
 - o לא ניתן להניח דבר על תוכן הרשימה. בפרט, טיפוס הנתונים לא ידוע מראש ולא חייב להיות זהה לכל איבר ברשימה. איברי הרשימה עשויים לחזור על עצמם.
-

דוגמאות:

```
is_palindrome([]): True
is_palindrome(['s']): True
is_palindrome([1,4,'g','g',4,1]): True
is_palindrome(['a','c','v']): False
```

3. בקורס הסתברות רוצים לבדוק את אלמנט המזל ברצף הטלות מטבע (כאשר בכל הטלה אנו מקבלים עץ או פלי). לצורך כך, הוחלט לכתוב תוכנה אשר בהינתן רצף הטלות מטבע תחשב מספר דברים:
- A. אחוז הפעמים שהוטל עץ (מספר הפעמים שהוטל עץ חלקי סך הכל ההטלות) כפול 100
- B. אחוז הפעמים שהוטל פאלי
- C. מספר הרצפים (הטלה מאותו הסוג) באורך 5 או יותר אשר נחשבים לברי מזל במיוחד. שימו לב שאם יש רצף באורך 7 לדוגמה, הוא נספר כרצף בר מזל אחד. המספר יהיה מסוג int.
- D. אורך הרצף הארוך ביותר. (ללא קשר עם התשובה של C) המספר יהיה מסוג int.
- כתבו פונקציה המבצעת את החישובים הללו על פי ההנחיות הבאות:

- הפונקציה מקבלת רשימה בה כל איבר תואם להטלה. אם יצא עץ, האיבר הוא 1, אחרת 0. לדוגמה, הרשימה [0,1,1,1,0] מייצגת הטלה של פאלי, עץ, עץ, עץ ולבסוף פאלי.
- חתימת הפונקציה:

def lucky_tosses(lst):

- הנחות על הקלט: רשימה ריקה היא חוקית ועדיין ניתן לבצע את החישובים הדרושים. במקרה זה, אחוז הפעמים שהוטל עץ/פאלי הוא 0.
- ניתן להניח שכל אברי הרשימה, אם היא לא ריקה, הם 0 או 1
- הפונקציה תחזיר רשימה של ארבעת הערכים A,B,C,D המתוארים לעיל, לפי הסדר. שימו לב שבתוך הפונקציה עליכם לתת לערכים אלו שמות טובים יותר.

דוגמה לשימוש בפונקציה:

עבור הרשימה: [1,1,0,0,1,1,1,1,1,1,0,0,0,0,1,1,1,0,0,0,0,0,1,1,1], הפונקציה תחזיר:
[57.69230769230769, 42.30769230769231, 2, 7]

עבור הרשימה [1,1,0,0,1,1,1,1,0,0,0,1,1,1], הפונקציה תחזיר
[64.28571428571429, 35.714285714285715, 0, 4]

חלק ג' – לולאות מקוננות

השאלות בפרק זה ניתנות לפתרון על ידי שימוש בלולאות מקוננות. עם זאת, אין חובה לקודד לולאות מקוננות "באופן מפורש" ובפרט אפשר מתוך לולאה לקרוא לפונקציה אחרת אשר מריצה בעצמה לולאה. כמו כן, לעתים קיים פתרון יעיל יותר (איך אין הכרח לממש אותו) ללא לולאה מקוננת.

4. התפלגות מצטברת: עבור רשימת מספר נתונה, פונקציית התפלגות מצטברת הוא פונקציה אשר עבור ערך a , מחזירה את החלק היחסי של המספרים אשר קטנים או שווים ל- a . לדוגמה, עבור הרשימה [4,3,2,1] וערך $a=3$ הפונקציה תחזיר 0.75 שכן שלושה מתוך ארבע האיברים קטנים או שווים ל-3. גם עבור $a=3.5$ הפונקציה תחזיר 0.75, אך עבור $a=2.5$ הפונקציה תחזיר 0.5. לפונקציית ההתפלגות המצטברת תפקיד מרכזי בתורת ההסתברות ועליכם לממש גרסה פשוטה של הפונקציה לפי ההנחיות הבאות:

- כתבו פונקציה המקבלת רשימת מספרים ורשימת ערכים ומחזירה רשימה ובה עבור כל אחד מהערכים חושב ערך ההתפלגות המצטברת שלו. לדוגמה, עבור רשימת המספרים [4,3,2,1] ורשימת הערכים [3.5, 2.0, -1], הפונקציה תחזיר שלושה ערכים התואמים ל $a=-1$, $a=2$, $a=3.5$, כלומר היא תחזיר [0.5, 0.75, 1.0].
- חתימת הפונקציה:

def cumulative_distribution(num_list, value_list):

- הנחות על הקלט: שתי הרשימות אינן ריקות ואינן חייבות להיות באותו אורך
- במימוש הפונקציה אין להשתמש ב: מודול numpy, בפונקציה count וב - collections.Counter.

-

דוגמה לשימוש בפונקציה

`cumulative_distribution ([1],[0]) → [0]`

`cumulative_distribution ([1],[1.5]) → [1]`

`cumulative_distribution ([1],[0, 1.5]) → [0, 1]`

`cumulative_distribution ([1, 2.2],[0, 1.5]) → [0, 0.5]`

`cumulative_distribution ([1, 2.2],[0, 1.5, 2.1, 2.2]) → [0, 0.5, 0.5, 1.0]`

`cumulative_distribution ([1, 2, 3, 4],[0, 0.5, 1, 2, 3, 4]) → [0, 0, 0.25, 0.5, 0.75, 1.0]`

5. זוגות שוות מכפלה: כתבו פונקציה המקבלת מספר שלם n , ומחזירה את רשימת כל הזוגות (רשימות באורך 2) של מספרים שלמים אשר מכפלתם שווה בדיוק ל- n מלבד הזוג הלא מעניין $[n, 1]$

- חתימת הפונקציה:

def equal_product_pairs(n):

- הנחות על הקלט: n הוא מספר שלם חיובי (int).
- במידה ואין אף זוג שמכפלתו n תחזיר הפונקציה רשימה ריקה. שימו לב שזה יקרה כאשר n ראשוני
- אין חשיבות לסדר הזוגות ברשימת הפלט ואין חשיבות לסדר האיברים בתוך כל זוג. לדוגמה, אם $n=6$ יש להחזיר את הזוג $[2,3]$ או $[3,2]$, אבל לא את שניהם
- אפשר להכפיל מספר עם עצמו
- שימו לב! הפונקציה חייבת להחזיר רשימות של מספרים שלמים בלבד! (int מסוג).

דוגמה לשימוש בפונקציה

`equal_product_pairs(5) → []`

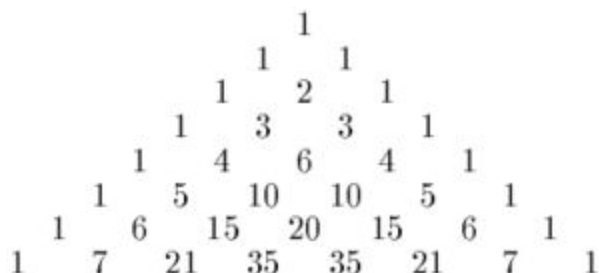
`equal_product_pairs(6) → [[3, 2]]`

`equal_product_pairs(9) → [[3, 3]]`

`equal_product_pairs(36) → [[2, 18], [3, 12], [4, 9], [6, 6]]`

הערה (אין חובה לממש): שימו לב שאפשר לממש את הפונקציה בצורה יעילה יחסית כך שניתן יהיה להפעיל אותה גם עבור n גדול מאוד ללא זמן המתנה ממושך. רמז: חשבו על איזה מספרים צריך לעבור?

6. משולש פסקל: משולש פסקל הוא משולש תמים למראה בו המספרים בנויים בצורת פירמידה וכל מספר בכל שורה שווה לסכום שני המספרים בשורה מעל שהוא נמצא ביניהם. האיבר הראשון והאחרון בכל שורה הוא 1 (ראה ציור):



באופן מפתיע, למשולש פסקל שימושים בתורת ההסתברות כפי שתלמדו בהמשך התואר.

- כתבו פונקציה המקבלת מספר שלם חיובי n ומחזירה רשימה המייצגת את משולש פסקל (ראה פרטים למטה)
- חתימת הפונקציה:

def pascal_triangle(n):

- ניתן להניח כי n הוא מספר שלם (int) גדול או שווה ל-1.
- הפונקציה תחזיר רשימה עם n איברים, כאשר כל איבר תואם לשורה במשולש פסקל.
- כל איבר יהיה בעצמו רשימה ויכיל את המספרים בשורה המתאימה למשולש.
- שימו לב! הפונקציה חייבת להחזיר רשימות של מספרים שלמים בלבד! (מסוג int).

לדוגמא:

- עבור $n=1$, הפונקציה תחזיר `[[1]]`
- עבור $n=2$, הפונקציה תחזיר `[[1], [1, 1]]`
- עבור $n=3$, הפונקציה תחזיר `[[1], [1, 1], [1, 2, 1]]`
- עבור $n=4$, הפונקציה תחזיר `[[1], [1, 1], [1, 2, 1], [1, 3, 3, 1]]`
- וכן הלאה

בדיקה עצמית של הקבצים: ראו הוראות במסמך "נוהל הרצת הטסטים" באתר הקורס.

הוראות הגשה

עליכם להגיש את הקובץ `ex3.zip` (בלבד) בקישור ההגשה של תרגיל 3 דרך אתר הקורס על ידי לחיצה על "Upload file".

`ex3.zip` צריך לכלול את הקבצים:

1. `ex3.py`
2. `README` (כמפורט בנהלי הקורס)

הנחיות כלליות בנוגע להגשה

- הנכם רשאים להגיש תרגילים דרך מערכת ההגשות באתר הקורס מספר רב של פעמים. ההגשה האחרונה בלבד היא זו שקובעת ושתיבדק.
- לאחר הגשת התרגיל, ניתן ומומלץ להוריד את התרגיל המוגש ולוודא כי הקבצים המוגשים הם אלו שהתכוונתם להגיש וכי הקוד עובד על פי ציפיותיכם.
- ניתן לאחר בהגשת התרגיל עד שלושה ימים, אך זה יהיה כרוך בניכוי ציון, כפי שמפורט באתר הקורס ובמסמך נהלי הקורס.
- קראו היטב את קובץ נהלי הקורס לגבי הנחיות נוספות להגשת התרגילים.
- שימו לב - יש להגיש את התרגילים בזמן!

בהצלחה!