

## תכנות הנדסי בשפת פייתון – תרגיל בית 1

### מתרגל אחראי: אופיר יעיש

#### הנחיות כלליות:

- מועד אחרון להגשה: כמפורסם בתיבת ההגשה ב-Moodle.
- מטרת התרגיל הינה לתרגל כתיבה, ולרכוש מיומנות בכלים שנלמדו עד כה בכיתה.
- קראו את העבודה מתחילתה ועד סופה לפני שאתם מתחילים לפתור אותה. ודאו שאתם מבינים את כל המשימות. רמת הקושי של המשימות אינה אחידה.
- את התרגיל יש לפתור לבד!
- בתיבת ההגשה במערכת ה-VPL, ישנו קובץ שלד לכתיבת הקודם שלכם. עבור כל משימה כתבו הפתרון שלכם במקום המתאים למשימה.
- אין למחוק שום קטע קוד הנמצא בשלד. עליכם רק להוסיף את הפתרון שלכם בתוכו.
- אין לשנות את שם/שמות ה-Requested files.
- השאלות יבדקו באופן אוטומטי. הפלט שעליכם להחזיר בכל תרגיל צריך להיות בדיוק כפי שנדרש. כמו כן, באופן אקראי יבדקו גם עבודות באופן ידני.
- כאשר תבוצע בדיקה ידנית, תתבצע גם בדיקת Readability - שימו לב שאתם משתמשים בשמות משתנים אינפורמטיביים וכותבים הערות בכל סעיף.
- בדיקה עצמית: כדי לוודא את נכונותן ואת עמידותן של הפונקציות לקליטים שונים, בכל שאלה הריצו אותן עם מגוון קליטים: אלה שמופיעים בדוגמאות וקליטים נוספים עליהם חשבתם. וודאו כי הפלט נכון. הבדיקה תתבצע על מגוון דוגמאות ולא בהכרח אלה שיינתנו פה.
- ניתן להשתמש בחומר הנלמד עד לפרסום העבודה ורק בחומר הזה.
- אין להשתמש בחבילות או מודולים חיצוניים (דוגמאת math), למעט מקרים שבהם צוין אחרת במפורש.
- במידה ולא צוין אחרת, יש להניח את נכונות הקלט על פי תיאור המשימה.
- על מנת לקבל ניקוד מלא, יש לענות נכונה על 3 מתוך 4 השאלות הראשונות, ועל השאלה מספר 5.
- משקל כל שאלה הוא זהה.
- במידה ולא עניתם על שאלה מסוימת, נא מלאו את הפונקציה בכל מקרה על מנת שהקוד שלכם יצליח לעבוד.

#### הנחיות חשובות למשימה זאת:

- בתרגיל זה, עליכם לבחור 3 מתוך ה-4 השאלות הראשונות לבדיקה. עליכם להכניס הבחירה שלכם למשתנה הגלובלי chosen\_questions מסוג רשימה הנמצא בשלד המוצע לכם. במידה והכנסתם יותר מ-3 שאלות ל-chosen\_questions, יבדקו שלושת השאלות הראשונות שמופיעות ברשימה.
- שאלה מספר 5 היא שאלת חובה.
- היצמדו להוראה של כתיבת פונקציות רקורסיבית במידה והתבקשתם לכך.
- במשימה זאת אתם רשאים לכתוב פונקציות עזר שיבצעו את התהליך הרקורסיבי כך שהפונקציות שהוגדרו בשאלה יהיו מעטפת לפונקציית העזר הרקורסיבית. בשאלות בהן מומלץ ואף הכרחי לעשות כך, רשמנו את האפשרות להוסיף פונקציית עזר כהנחיה.
- לאורך כל העבודה, חל איסור על שימוש ב-built-in methods של string ו-list למעט הפעלת len ו-append. ניתן להשתמש ב-splicing ו-indexing. אם אתם חושבים שפונקציה מסוימת הכרחית לפתרון שלכם, ניתן לשאול בפורום ונשקול לאשר את השימוש בה.
- חל איסור על שימוש במשתנים גלובליים, למעט השימוש במשתנה הגלובלי chosen\_question לבחירת שלושת השאלות לבדיקה.

## שאלה 1

ממשו את הפונקציה הרקורסיבית `identity_except_one(str1, str2)` המקבלת 2 מחרוזות ומחזירה את הערך הבוליאני True אם ורק אם הן שוות בתויהם פרט למיקום אחד בדיוק. במידה והמחרוזות באורך שונה, על הפונקציה להחזיר את הערך הבוליאני False.

### דוגמאות:

עבור הקריאה `identity_except_one("001", "011")` יוחזר הערך True.

עבור הקריאה `identity_except_one("011", "011")` יוחזר הערך False.

עבור הקריאה `identity_except_one("011", "0110")` יוחזר הערך False.

## שאלה 2

**תת-קבוצה סדורה (הגדרה):** בהינתן 2 מחרוזות `str1` ו-`str2`, נאמר כי `str2` היא תת-קבוצה סדורה של `str1` אם `str1` מכילה את כל התווים של `str2` באותו הסדר בו הם מופיעים ב-`str1`. לדוגמה מחרוזת 'abc' היא תת-קבוצה סדורה של 'ladbcbfe', אבל אינה תת-קבוצה סדורה של 'lbdacfe' כיוון שהתווים 'a' ו-'b' אינם מופיעים באותו הסדר במחרוזות 'abc' ו-'lbdacfe'.

ממשו את הפונקציה הרקורסיבית `ordered_subset(str1, str2)` המקבלת 2 מחרוזות `str1` ו-`str2`, ומחזירה את הערך הבוליאני True אם `str2` היא תת-קבוצה סדורה של `str1` **כך שכל 2 תווים עוקבים ב-`str2` אינם תווים עוקבים ב-`str1`.** אחרת, יוחזר הערך הבוליאני False. הניחו כי `str1` אינה ריקה **ומכילה תווים שונים.**

### דוגמאות:

עבור הקריאה `ordered_subset("ladbcbfe", "abc")` יוחזר False.

עבור הקריאה `ordered_subset("ladbxcfe", "abc")` יוחזר True.

## שאלה 3

ממשו את הפונקציה `k_size_subsets(n, k)` המקבלת 2 מספרים שלמים  $n$  ו- $k$ , ומחזירה כרשימה את כל תתי הקבוצה בגודל  $k$  של הקבוצה  $\{1, 2, 3, \dots, n\}$  בסדר כלשהו. כל תת-קבוצה ברשימה המחוזרת תיוצג ע"י מחרוזת של איברי תת הקבוצה. על כל תווי המחרוזת המייצגת תת-קבוצה להיות בסדר עולה משמאל לימין. ניתן להניח כי  $0 \leq k \leq n$ .

**הנחיה:** חובה להגדיר פונקציית עזר רקורסיבית, ובה להגדיר פרמטרים כרצונכם.

### דוגמאות:

עבור הקריאה `k_size_subsets(5, 3)` תוחזר הרשימה (עד כדי שינוי בסדר המחרוזות המייצגות תתי-קבוצות):

`['123', '124', '125', '134', '135', '145', '234', '235', '245', '345']`

עבור הקריאה `k_size_subsets(5, 0)` תוחזר הרשימה עם המחרוזת הריקה:

[“]

**שימו לב:** עבור כל מחרוזת, על תווי המחרוזת להיות בסדר עולה. לכן לדוגמה, המחרוזת '342' שמייצגת את תת הקבוצה {2,3,4} לא תתקבל, והמחרוזת המייצגת שתתקבל היא '234'.

#### **שאלה 4:**

ממשו את הפונקציה הרקורסיבית  $\text{linear\_sum}(x, \text{result})$  המקבלת רשימה  $x = [x_0, x_1, \dots, x_{n-1}]$  של מספרים שלמים וערך שלם  $\text{result}$ , ומחזירה את הערך הבוליאני True אם קיימים מקדמים  $a_i \in \{-1, 0, 1\}$  עבור  $0 \leq i \leq n-1$  כך שהצירוף הלינארי הבא  $a_0x_0 + a_1x_1 + \dots + a_{n-1}x_{n-1}$  שווה בדיוק ל- $\text{result}$ . אחרת הפונקציה תחזיר את הערך הבוליאני False.

#### **דוגמאות:**

עבור הקריאה  $\text{linear\_sum}([2,3,6,7,10], 15)$  יוחזר הערך True שכן:







$$1 * 2 + (-1) * 3 + 1 * 6 + 0 * 7 + 1 * 10 = 15$$

עבור הקריאה  $\text{linear\_sum}([5, 14, 7, 3], 20)$  יוחזר הערך False שכן לא קיים צירוף לינארי כנ"ל שסכומו 20.

#### **שאלה 5:**

בשאלה זו נפתור את בעיית ח-המגדלים. בבעיה נתון לוח ריבועי. מספרי השורות והעמודות בלוח נספרים מ-0. על כל משבצת יכול להיות מגדל יחיד. מרחק בין שני מגדלים הוא מספר ההזזות המינימלי שנדרש לבצע על מנת להזיז מגדל אחד למקומו של השני. **הזזה אחת של מגדל תוגדר להיות הזזה של המגדל לאחת מארבע המשבצות הסמוכות לו: מעל, מתחת, מימין ומשמאל (לא ניתן להזיז את המגדל באלכסון).**

לדוגמה, המרחק בין מגדל F למגדל E באיור הבא הוא 4 כיוון שנדרשים 4 צעדים כדי להגיד ממשבצת (5, 5) למשבצת (4, 2).

	0	1	2	3	4	5
0	 A					
1			 B			
2					 C	
3	 D					
4			 E			
5						 F

בבעיית n-המגדלים נתון מספר שלם לא שלילי d ולוח ריבועי בעל n שורות ועמודות, ועלינו להציב n מגדלים כך ש:

- (1) בכל שורה יהיה מגדל אחד בלבד.
- (2) המרחק בין כל שני מגדלים יהיה גדול ממש מ-d.

נקרא ל-d "מרחק סף"

לדוגמה, עבור d=2, הלוח בתמונה לעיל מציגה פתרון לבעיה.

**ייצוג הלוח ומיקום המגדלים בשאלה:** נייצג לוח ריבועי בגודל nXn בעזרת רשימת באורך n. הערך באינדקס i של הרשימה מייצג את העמודה שבה מוצב המגדל בשורה ה-i-ית בלוח.

לדוגמה, הלוח באיור לעיל ייוצג על ידי הרשימה הבאה:

board = [0, 2, 4, 0, 2, 5]

מטרת סעיפי השאלה לחלק את פתרון הבעיה לתת משימות קטנות יותר:

### סעיף א':

ממשו את הפונקציה distance(row1, col1, row2, col2) המחשבת את המרחק בין שני מגדלים המוצבים במשבצות (row1, col1) ו-(row2, col2). ניתן להניח כי הקלט תקין.

הערה: המימוש כאן אינו חייב להיות רקורסיבי.

### דוגמה:

עבור הקריאה distance(5, 5, 4, 2) יוחזר הערך 4.

### סעיף ב':

ממשו את הפונקציה add\_tower(board, d, row, col) המקבלת רשימה board המייצגת את הלוח, מספר d המייצג את מרחק הסף, ושני מספרים המייצגים שורה row ועמודה col בלוח. הפונקציה תחזיר את הערך הבוליאני True

אם ניתן להציב במשבצת (row, col) מגדל כך שהמרחק שלו מכל מגדל הנמצא באחת השורות מעליו יהיה גדול ממש ממרחק הסף d. כמו כן, במידה וניתן להציב את המגדל, הפונקציה תציב את המגדל בלוח. כלומר, הפונקציה תציב את הערך col בתא row של הרשימה board. אם לא ניתן להציב מגדל, על הפונקציה להחזיר את הערך הבוליאני False מבלי לשנות את הרשימה.

ניתן להניח כי הקלט תקין.

הערה: המימוש כאן אינו חייב להיות רקורסיבי.

### דוגמאות:

בהינתן הלוח שמיוצג על ידי:

```
board = [0, 3, 5, 0, 0, 0]
```

עבור הקריאה add\_tower(board, 2, 3, 3) יוחזר הערך False כי המשבצת (3,3) נמצאת במרחק 2 מהמשבצת (1,3) הנמצאת שתי שורות מעליו, ולכן מרחק זה אינו גדול ממש מ-d=2. כמו כן, הרשימה board לא תשתנה.

עבור הקריאה add\_tower(board, 2, 3, 1) יוחזר הערך True. כמו כן, תשתנה הרשימה ל-

```
board = [0, 3, 5, 1, 0, 0]
```

### סעיף ג':

ממשו את הפונקציה n\_towers(n, d) המקבלת גודל לוח ריבועי n ומרחק סף d. הפונקציה תחזיר רשימה באורך n המייצגת פתרון כלשהו לבעיית n-המגדלים עם מרחק סף d. אם לא קיים פתרון, הפונקציה תחזיר רשימה ריקה. ניתן להניח כי הקלט תקין.

הנחיה: חובה להגדיר פונקציה עזר רקורסיבית, ובה להגדיר פרמטרים כרצונכם.

לשם פשטות, ניתן להגדיר לוח התחלתי עם n מגדלים הנמצאים בעמודה שבאינדקס 0. כלומר להגדיר רשימה התחלתית המייצגת לוח עם אפסים.

### דוגמאות:

1. עבור הקריאה n\_towers(6, 2), רשימה אפשרית שתוחזר היא:

```
[0, 2, 4, 0, 2, 4]
```

2. עבור הקריאה n\_towers(6, 6) תוחזר הרשימה הריקה:

```
[]
```