



# דוח משוב לסטודנט

מזהה סטודנט: N102447804  
 תאריך בחינה: ים שלישי, 1 ביולי 2025  
 מזהה קורס: 20606  
 תכנות וiyuth נתונם במשפט פיתון  
 שם עבוזי: מנכלה  
 מרצה:

ציון מבחן סופי

ציון מבחן מקורי

ניקוד שאלות פתוחות

**87.00****87.00****87.00**

## סיכום

מספר שאלת	תיאור	נקוד	נקוד מרבי	ניקוד מרבי
1	שאלת 1	20.00	25.00	
2	שאלת 2	25.00	25.00	
3.1	סעיף א	12.00	12.00	
3.2	סעיף ב	10.00	13.00	
4	שאלת 4		25.00	
5.1	סעיף א	5.00	5.00	
5.2	סעיף ב	3.00	3.00	
5.3	סעיף ג	4.00	4.00	
5.4	סעיף ד	6.00	8.00	
5.5	סעיף ה	2.00	5.00	

# מוס' שאלון - 527

1

ביזייל 2025

83 מועד מוס'

סמסטר 2025 ב  
20606 / 4

## שאלון בחינת גמר

### 20606 - תוכנות וניתוח נתוניים במשפט פיטון

משך בחינה: 3 שעות

בשאלון זה 22 עמודים

#### מבנה הבחינה:

קראו בעיון את ההוראות שלහן:

- \* בבחינה יש חמישה שאלות. עליכם לענות על ארבע מתוכן על גבי שאלון הבחינה, על פי סמסטר הלימוד שלכם.
- \* יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה.
- \* תכנית לא יעילה לא תקבל את מלאן הנקודות.
- \* אם ברצונכם להשתמש בפונקציה המוגדרת בשפה ניתן להניח כי המודול הרלוונטי מזובא כנדרש. יחד עם זאת, יש להקפיד על זימון הפונקציה באופן מדויק (פרמטרים, טיפוס פרמטרים וכו').
- \* אין להשתמש במודולים ופונקציות קיימות בשפה, חוץ מآلן אשר המופיעים בחומר הלימוד של הקורס ובהתאם להנחיות המפורטות בשאלות הבחינה.
- \* יש לשמר על סדר. תכנית הכתובה בצורה בלתי מסודרת עלולה לגרום מהצין.
- \* בכל השאלה ניתן להניח כי הקלט תקין, אלא אם כן מצוין אחרת.
- \* בכתיבת התכניות יש להשתמש רק במרכיבי השפה שנלמדו בקורס זה.
- \* השלימו את הדרוש, במקומות מסוימים בקווים. מספר השורות הריקות לא זהה בהכרח למספר השורות בפתרון.
- \* שימו לב, כל התשובות לשאלות צריכות להכתב על גבי שאלון הבחינה.

תשובה שתיכתב במקום אחר לא תיבדק!

#### חומר עזר:

כל חומר עזר מודפס ו/או כתוב בכתב יד, מותר בשימוש.  
אסור שימוש בעזרים דיגיטליים, ובחומרים מוקוונים.

הצדiron  
למשגיח את השאלון  
וכל עזר אחר שקיבלתם בתור מחברת התשובות

בצלחה !!!





## קראו בעיון את ההוראות שלහלן:

בבחינה יש חמיש שאלות:

- שאלה 3-1 הן חובה לכל הסטודנטים.
  - שאלה 4 מיועדת לסטודנטים אשר למדו את הקורס בשנת הלימודים 2024.
  - שאלה 5 מיועדת לסטודנטים אשר לומדים את הקורס בשנת הלימודים 2025. בסך הכל, עלייכם לענות על ארבע מתחום על גבי שאלון הבחינה. יש לקרוא היטב את ההוראות לمعנה על שאלות בהתאם לזאותכם.
1. יש להזכיר בכתב את התכניות בצורה מבנית ויעילה. **תכנית לא יעילה לא תקבל את מלאה הנקודות.**
2. אם ברצונכם להשתמש בפונקציה המוגדרת בשפה ניתן להניח כי המודול הרלוונטי מיובא-CNDRSH. יחד עם זאת, יש להזכיר על זימנון הפונקציה באופן מדויק (פרמטרים, טיפוסי נתונים וכו').
3. יש להשתמש במודולים ופונקציות הקיימות בשפה אשר הינם חלק מהחומר הלימוד של הקורס ובהתאם להנחיות המפורחות בשאלות הבחינה.
4. יש לשמר על סדר. **תכנית הכתובה בצורה בלתי מסודרת עלולה לגרוע מהציון.**
5. בכל שאלות ניתן להניח כי הקלט תקין, אלא אם כן מצוין אחרת.
6. בכתב התכניות יש להשתמש רק במרכיבי השפה שנלמדו בקורס זה.
7. השלימו את הדריש, **במקומות מסווגים בקווים. מספר השורות הריקות לא זהה בהכרח למספר השורות בפתרון.**

**שים לב, כל התשובות לשאלות צריכות להכתב על גבי שאלון הבחינה. תשובה שתיכתב במקום אחר לא תיבדק!**

**חומר עזר המותר בשימוש הוא**

**כל חומר כתוב בכתב יד או מודפס**

**אסור להשתמש במחשב מכל סוג שהוא!**



### שאלה 1 (25 נקודות)

נאמר שchorה של מספרים מייצגים גבהים של נקודות ברכס, שיש בהם פסגות ועמקים. כתבו פונקציה בשם max\_drop המתקבלת רשימה לא ריקה *lst* של מספרים חיוביים. הפונקציה תחזיר את הנפילה המקסימלי בין שני מספרים (לא בהכרח במקומות סמוכים), כך שהמספר הגבוה נמצא לפני המספר הנמוך ברשימה ותדפס את ערכיו איברים אלו. הרשימה *lst* אינה ממוינת. יש לפטור את הבעיה בסדר גודל לינארי ביחס לרשימה *lst* תוך מעבר ייחיד על הרשימה. פתרו בסיבוכיות ריבועית, יזכה לניקוד חלקי בלבד.

דוגמאות:

- עבור הרשימה [4, 5, 21, 3, 27, 12, 24, 7, 6, 23]. יודפסו האיברים 4, 27, 12, 24, 7, 6, 4 ויווחזר הערך 23.
- עבור הרשימה [14, 5, 21, 3, 22, 12, 7, 26, 18]. יודפסו האיברים 21, 3, 22, 12, 7, 26, 14 ויווחזר הערך 18.
- עבור הרשימה [5, 15, 3, 22, 12, 7, 27, 14]. יודפסו האיברים 22, 12, 7, 27, 14 ויווחזר הערך 15.

הערות:

- ניתן להניח ש- *lst* מייצג אובייקט מסווג רשימה (list) לא ריקה המכילה מספרים חיוביים.
- אין להשתמש ברשימות נוספות או מבני נתונים אחרים אלא לסרוק את הרשימה *lst*.

התשובה היא:

*def max\_drop(lst)*

-2

שם ת מעתנים לא שימושיים

(1)

*for i in range(len(lst)):*

*if lst[i] > lst[i+1]:*

*i = j*

*else:*

*res = max(res, lst[i] - lst[i+1])*

8.2

*res = res + lst[i]*

-3

(1)

אין הדפסה של הערכים

*print(res)*

*print(max\_drop([1, 2, 3, 4, 5, 6, 7, 8, 9, 10]))*

*27*

*27*



0(1)  $\sim$   $O(n) - O(n)$   
,  $n$   $\sim$   $n$   $\sim$   $n$

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

if  $1st[i] - 1st[j] > hes$ :

$hes = 1st[i] - 1st[j]$



## שאלה 2 (25 נקודות)

הרשימה `lst` מלאה במספרים שלמים חיוביים הממוינים בסדר עולה ממש (כלומר אין ערכים החוזרים על עצם).

כתבו פונקציה בשם `count_triples` המקבלת כפרמטרים את הרשימה הממוינה `lst` ומספר שלם `num`. הפונקציה תחזיר את מספר שלשות המספרים בראשימה `lst` שהמכפלה שליהם היא `num`. הפונקציה צריכה להיות עיליה בהיבט סיבוכיות זמן ריצה ככל הניתן. פתרו בסיבוכיות גדולה מסיבוכיות ריבועית תזוכה בניקוד חלקי.

**דוגמאות:**

- נתונה הרשימה `[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 14, 18, 19, 22]`.
- עבור `num=60` יוחזר הערך 3 עבור השלשות `2, 3, 10` ו- `3, 4, 5`.
- עבור `num=80` יוחזר הערך 2 עבור השלשות `2, 4, 10` ו- `3, 5, 8`.
- עבור `num=20` יוחזר הערך 0. אין אף שלשה שמכפלה שווה ל- 20.

**הערות:**

- ניתן להניח ש- `lst` מייצגת אובייקט מסווג רשימה (list) לא ריקה המכילה מספרים שלמים חיוביים הממוינים בסדר עולה ממש.
- אין להשתמש ברשימות נוספות או מבני נתונים אחרים אלא לסרוק את הרשימה `lst`.

התשובה היא:

25  
(2)

$$\text{def count\_triples(lst):}$$

$$\text{ res = 0}$$

$$\text{ for i in range(len(lst)):$$

$$\text{ for j in range(i+1, len(lst)):$$

$$\text{ for k in range(j+1, len(lst)):$$

$$\text{ if lst[i] * lst[j] * lst[k] == target:}$$

$$\text{ res += 1}$$



elif  $1st[k] + 1st[j] > target$ :

$j = j - 1$  # 'yi niv p'nei

else:

$k = k + 1$  # 'yli niv p'nei

break res

time  $\Rightarrow$   $O(1) \sim O(N)$   $\rightarrow$   $O(1)$

space  $\Rightarrow$   $O(1)$   $\rightarrow$   $O(1)$

algorithm  $\rightarrow O(h^2) = O(N^2)$   $\rightarrow$   $O(N^2)$

complexity  $\rightarrow O(N^2)$   $\rightarrow$   $O(N^2)$   $\rightarrow$   $O(N^2)$

time  $\rightarrow$   $O(N^2)$   $\rightarrow$   $O(N^2)$

, two pointers  $\Rightarrow$   $O(N)$   $\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$

space  $\rightarrow$   $O(1)$   $\rightarrow$   $O(1)$   $\rightarrow$   $O(1)$

$\rightarrow$   $O(\frac{h^2}{N})$   $\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$

complexity  $\rightarrow O(N)$   $\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$

$\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$   $\rightarrow$   $O(N)$

space  $\rightarrow O(1)$   $\rightarrow O(1)$   $\rightarrow O(1)$

, time  $\rightarrow O(N)$   $\rightarrow O(N)$   $\rightarrow O(N)$

, coherency  $\rightarrow O(N)$   $\rightarrow O(N)$   $\rightarrow O(N)$

$\rightarrow O(N)$

if  $ham > 1st[i] \neq 0$ :

coherency

$\rightarrow O(N)$   $\rightarrow O(N)$   $\rightarrow O(N)$



### שאלה 3 (25 נקודות)

#### סעיף א: (12 נקודות)

כתבו פונקציה רקורסיבית בשם `is_sub_list_non_seq` המקבלת שתי רשימות מספרים `lst`, `sub`.  
 הפונקציה תחזיר `True` אם כל איברי הרשימה `sub` נמצאים ברשימה `lst` לא בהכרח ברצף אך באותו הסדר. אחרת, יוחזר `False`.

דוגמאות:

- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10]`, `sub=[4, 10]` יוחזר `True`.
- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10]`, `sub=[2, 3, 10]` יוחזר `True`.
- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10]`, `sub=[3, 1, 10]` יוחזר `False`. אומנם כל איברי הרשימה `sub` נמצאים ברשימה `lst` אך לא בסדר הופעתם ברשימה `sub`.
- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10]`, `sub=[5, 8, 1, 11]` יוחזר `False`. הערך `11` לא נמצא ברשימה `lst`.
- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10]`, `sub=[]` יוחזר `True`.

הערות:

- פונקציה שאינה רקורסיבית לא תזכה בנקודות כלל.
- ניתן להניח שהרשימות `sub`, `lst` מייצגות אובייקטים מסווג רשימה (list) המכילות מספרים.
- אסור לכתוב פונקציות עזר ו/או לעשות שימוש בהעמסת פרמטרים עם ערכי בריית מחדל.
- אין לעשות שימוש בולולאות כלל.

התשובה היא:

```

def is_sub_list_non_seq(lst, sub):
    if not sub:  # אין מילים
        return True
    if not lst:  # אין מילים
        return False
    if sub[0] == lst[0]:  # מילים שווים
        return is_sub_list_non_seq(lst[1:], sub[1:])
    else:  # מילים לא שווים
        return is_sub_list_non_seq(lst[1:], sub)

```



### --- המשך השאלה בעמוד הבא ---



**סעיף ב: (13 נקודות)**

הfonקציה `True` אם כל איברי הרשימה `sub` נמצאים בראצף בראשימה `lst`. אחרת, יוחזר `False`.

## דוגמאות:

- עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10], sub=[4, 10]`, `True` יוחזר
  - עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10], sub=[2, 4, 10]`, `False` יוחזר. אומנם כל איברי הרשימה `sub` נמצאים ברשימה `lst` אך לא בראצ'.
  - עבור הרשימות `lst=[5, 2, 8, 1, 3, 4, 10], sub=[ ]`, `True` יוחזר

## הערות:

- פונקציה שאינה רקורסיבית לא תזכה בנקודות כלל.
  - ניתן להנח שהרשימות sub, *lst* מייצגות אובייקטים מסווג רשימה (list) המכילים מספרים.
  - ניתן לכתוב פונקציות עזר ו/או לעשות שימוש בהעמסת פרמטרים עם ערכי ברירת מחדל.
  - אין לעשות שימוש באופרטור *in* המוגדר בשפה.
  - אין לעשות שימוש בולולאות כלל.

### ה תשובה היא:

```

if is_sub_list_seq(1st, sub):
    if not 1st: # 空のリスト
        return False
    if fibs_set(1st, sub, 0): # 1st が sub の初期部分
        return True
    return is_sub_list_seq(1st[1:], sub)
# finds a consecutive sequence in a list
def fibs_set(1st, sub, i):
    if not sub:
        return True
    if i >= len(1st) or 1st[i] != sub[0]:
        return False
    return fibs_set(1st, sub[1:], i+1)

```

מה קורה אם האיבר הראשוני בסיס מופיע פעמיים, והרצף מתקיים עם המופיע השני?



הנתקין	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן
הנתקן	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן
הנתקן	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן	הנתקן



לפניך שתי שאלות. יש לענות על שאלה **אחד מבין השתיים** בהתאם לסטודנט הלימוד שלך.

• אם למדת את הקורס בסמסטר 2024 א' או 2024 ב' – חובה לענות על שאלה 4.

• אם למדת את הקורס בסמסטר 2025 א' או 2025 ב' – חובה לענות על שאלה 5.

**لتשומת לך!** באחריותך להקפיד על בחירת השאלה המתאימה.

#### שאלה 4 (25 נקודות) - רק למי שלמד/ה את הקורס בסמסטר 2024 א/ב

פיקסל (Pixel) הוא ייחידת מידע המתארת נקודה בתמונה דיגיטלית.

כל פיקסל מורכב משילוב של שלושה צבעי יסוד: אדום, ירוק וכחול.

כל אחד משלושת הצבעים האלה (אדום, ירוק וכחול) מקבל ערך מספרי מ- 0 עד 255 (כולל). המספר

מייצג את עוצמת הצבע בנקודה. באמצעותו שילוב של שלושת הצבעים בערכים שונים מ- 0 עד 255

אפשר לקבל כל גוון במרחב הצבעים האפשרי.

הצבע השחור נוצר באמצעותו עוצמת גוון 0 בכל אחד משלושת צבעי היסוד. לעומת זאת, הצביע הלבן

נוצר באמצעותו עוצמת גוון 255 בכל אחד משלושת צבעי היסוד.

המחלקה **Pixel** מייצגת פיקסל, ולה שלוש תכונות: אדום - red, ירוק - green וכחול - blue.

לקבלת נקודה שהצבע שלו הוא אדום (בין אדום בהיר לאדום כהה), ערכי הצבעים צריכים להיות:

אדום – מ- 1 עד 255, ירוק – 0, כחול – 0.

לקבלת נקודה שהצבע שלו הוא ירוק (בין ירוק בהיר לירוק כהה), ערכי הצבעים צריכים להיות:

אדום – 0, ירוק – מ- 1 עד 255, כחול – 0.

לקבלת נקודה שהצבע שלו הוא כחול (בין כחול בהיר לכחול כהה), ערכי הצבעים צריכים להיות:

אדום – 0, ירוק – 0, כחול – מ- 1 עד 255.

לקבלת נקודות בשאר הצבעים (לא כולל שחור ולבן), הערכים של לפחות שני צבעים מותוק צבעי

היסוד אדום, ירוק וכחול צריכים להיות גדולים מ- 0.

לקבלת נקודה שהצבע שלו הוא שחור, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 0.

לקבלת נקודה שהצבע שלו הוא לבן, ערכי הצבעים אדום, ירוק וכחול צריכים להיות 255.

--- המשך השאלה בעמוד הבא ---



במחלקה **Pixel** הוגדרו הפונקציות `is_red`, `is_green`, `is_blue` המחזירה `True` אם האובייקט עליו מופעלת הפונקציה מייצג צבע אדום, ירוק, כחול, בהתאם. אחרת, תחזיר `False`. הוגדרה גם הפונקציה `average` המחזירה את ממוצע הפיקסל באמצעות חישוב ממוצע של שלושת גווני צבעי היסוד. בנוסף, הוגדרו פונקציות `get` לכל אחת מהתכונות. אין צורך למשם פונקציות אלו!  
משתני המופיע בכל מחלוקת לא יגדרו ציבוריים אלא כמורנים (`protected`) באמצעות מקף תחתון, כלומר `.self._attribute_name`.

בסעיפים הבאים השלימו את הדריש, **במקומות שימושניים בקווים**. מספר השורות הריקות לא זהה בהכרח למספר השורות בפתרון.

### סעיף א: (4 נקודות)

כתבו במחלקה Pixel פונקציית בנאי המקבלת ערכי צבעי היסוד: אדום, ירוק, כחול. במידה וערך אחד או יותר מצבעי היסוד אינם חוקי (טוווח ערכים 0-255), יש לאותחל את ערך התכונה ל-1.

```
def __init__(self, red, green, blue):
```



**סעיף ב: (4 נקודות)**

כתבו במחלקה Pixel פונקציה בשם `invert` אשר הופכת גוון של כל צבע יסוד. היפוך גוון נעשה באמצעות הערך המשלים ל- 255 עבור כל צבע יסוד בפיקסל. למשל, אם ערכי הפיקסל הם : אדום : 240, ירוק : 15, כחול : 50. ערכי הפיקסל לאחר היפוך יהיו אדום : 55, ירוק : 15, וכחול : 205.

### חתימת הפונציה:

```
def invert (self):
```

### --- המשך השאלה בעמוד הבא ---



### סעיף ג: (5 נקודות)

כתבו במחלקה `Pixel` פונקציה המתקבלת תוספת גוון `amount` מטיפוס `Slm`. הפונקציה תחזיר אובייקט `Pixel` חדש שערך צבעי היסוד זהים לאובייקט הנוכחי בתוספת `amount` לכל אחד מצבעי היסוד של הפיקסל החדש. אין לשנות את ערכי האובייקט הנוכחי. הניחו כי ערך התוספת הוא מספר אי-שלילי (0 או חיובי). במקרה של חריגה בטוחה של אחד (או יותר) מצבעי היסוד, יש לעדכן את ערך הצבע התואם לערך המשלים - 255. למשל, אם גוון הצבע האדום הוא 240 והעדכוו הוא 25 אז ערכו לאחר העדכון באובייקט החדש יהיה 265 אך הוא יעדכן ל- 10. יש לעדכן את כל צבעי היסוד: אדום, ירוק וכחול עם התוספת `amount` בפיקסל החדש.

### חתימת הפונציה:

```
def updated_pixel (self, amount):
```



המחלקה **Image** מייצגת תמונה ולה שתי תכונות: שם התמונה - name המיצגת מחרוזת ורשימת פיקסלים המרכיבים את התמונה, xls.

במחלקה **Image** הוגדרו פונקציות `get` לכל אחת מהתכונות. אין צורך למשתמש פונקציות אלו!  
משתני המופיע בכל מחלוקת לא יגדרו ציבוריים אלא כמורנים (`protected`) באמצעות מקף תחתון,  
כלומר `.self. attribute name`

#### סעיף ד: (6 נקודות)

כתבו במחלקה `Image` פונקציה `rgb_mapping` המחזירה רשימה סדורה (tuple) של כמה הפיקסלים הקיימים בתמונה שהם צבעי יסוד. תזכורת: פיקסל בצבע יסוד הינו פיקסל בו ערך צבע היסוד הוא בטוחה 1 והוגונים של יתר צבעי היסוד הם 0. למשל, פיקסל אדום מיוצג ע"י ערבי הגוגנים: אדום : 1-255, ירוק : 0-255, כחול : 0. סדר החזרת הרשימה הסדרה הוא:

למשל, עבור רשימת הפיקסלים `pxls` הבאה:

```
pxls = [(255, 0, 0), (240, 10, 255), (0, 0, 255), (255, 10, 150), (0, 0, 255)]
```

תוחזר הרשימה הסדורה (tuple) הבהה : ( 2, 0, 1 ), כלומר פיקסל אחד בצבע אדום, אין כלל פיקסלים בצבע יירוק ושני פיקסלים בצבע כחול.

### חתימת השיטה:

```
def rgb_mapping (self):
```



#### סעיף ה: (6 נקודות)

כתבו במחלקה `Image` פונקציה בשם `is_brighter` שמקבלת תמונה נוספת. הפונקציה תחזיר `True` אם התמונה עליה מופעלת הפונקציה בהירה יותר מהתמונה הנוספת. אחרת, תחזיר `False`. תמונה בהירה יותר הינה תמונה שסכום ערכי הפיקסלים שלה גבוהה יותר. הנינו שהפרמטר המתקבל מיציג אובייקט מטיפוס `Image`.

### חתימת הפונציה:

```
def brighter (self, other):
```



## שאלה 5 (25 נקודות) - רק למי שLEARNS את הקורס בסמסטר 2025/2025 ב

כדי לנהל את מערכת האירועים שלה, החלטת הנהלת האוניברסיטה, להגדיר את המחלקות הבאות:

המחלקה Date המייצגת תאריך, שלה שלוש תוכנות המייצגות ערכיים שלמים: יום – day, חודש –

year ו שנה – month

בנוסף, נתונה פונקציית הבנייה הבאה למחלקה:

```
def __init__(self, day, month, year):  
    self._day = day  
    self._month = month  
    self._year = year
```

לצורך ניהול האירועים נבנתה המחלקה Event המייצגת אירוע.

במערכת האירועים קיימים 3 סוגי אירועים: פגישות, שיחות טלפון ומשימות. כדי לשמר את המידע שבכל סוג אירוע, הוגדרו המחלקות הבאות: פגישה – Meeting, שיחת טלפון –

משימה – Task.

להלן פירוט התוכנות של המחלקות, בהתאם לדרישות המערכת:

המידע הנשמר על המחלקה אירוע – date

hour – המייצגת שעת התחלתה, כמספר שלם בטווח 8-20

המידע הנשמר על המחלקה פגישה – Meeting

בנוסף לתוכנות של המחלקה Event הוגדרו התוכנות הבאות:

names – המייצגת רשימה של שמות המשתתפים בפגישה, כל שם מייצג מחוوظה

duration – המייצגת את משך הפגישה, כמספר שלם וחובי

location – המייצגת את מקום הפגישה, כמחוوظה

המידע הנשמר על המחלקה שיחת טלפון – PhoneCall

בנוסף לתוכנות של המחלקה Event הוגדרו התוכנות הבאות:

phone\_number – המייצגת מספר הטלפון שמחזגים, כמחוوظה

name – המייצגת שם האדם שאליו מתקשרים, כמחוوظה

--- המשך השאלה בעמוד הבא --



### המידע הנשמר על המחלקה משימה – Task

בנוסף לתוכנות של המחלקה Event הוגדרו התוכונה הבאה:  
title – המיצגת כוותרת המשימה, כמחuzeות

משתני המופיע בכל מחלקה לא יגדרו ציבורריים אלא כמוגנים (protected) באמצעות מקף תחתיו,  
.self.\_attribute\_name

בסעיפים הבאים השlimו את הדריש, **במקומות שמסומנים בקווים**. **מספר השירותות הריקות לא זהה בהכרח למספר השירות בפתרון.**

**סעיף א: (5 נקודות)**  
עבור כל מחלקה המתוארת בשאלת, יש לכתוב את כוותרת המחלקה (שורה ראשונה בהגדרת המחלקה). הנירו כי כל המחלקות נכתבות באותו קובץ.  
התשובה היא:

class Date;

---

+5

class Event(5.1)

---

class Meeting(Event):

---

class Phonecall(Event):

---

class Task(Event):

---

--- המשך השאלה בעמוד הבא ---



**סעיף ב: (3 נקודות)**

כתבו פונקציית בניאי במחלקה אירוע (Event) המתקבלת את שעת ההתחלה ותאריך אירוע  
באמצעות מרכיבי התאריך: יום, חודש ו שנה. ניתן להניח כי ערכי הפרמטרים המתקבלים תואימים  
את טיפוס הנתונים שלהם.

**התשובה היא:**

```
def __init__(self, hour+3, day, month, year):  
    self.(5.2) date = Date(day, month, year)  
    self.(5.3) hour = hour
```

**סעיף ג: (4 נקודות)**

כתבו פונקציית בניאי במחלקה שיחת טלפון (PhoneCall) המתקבלת את שעת ההתחלה, תאריך  
האירוע באמצעות מרכיבי התאריך: יום, חודש ו שנה, מספר הטלפון אליו מחייגים ושם האדם אליו  
מחייגים. ניתן להניח כי ערכי הפרמטרים המתקבלים תואימים את טיפוס הנתונים שלהם.

**התשובה היא:**

```
def __init__(self, hour, day, month, year, phone_number, name):  
    self.(5.4) date = Date(hour, day, month, year)  
    self.(5.3) phone_number = phone_number  
    self.(5.3) name = name
```

--- המשך השאלה בעמוד הבא ---



#### סעיף ד: (8 נקודות)

ברצוננו למשם את הפונקציה `match` אשר מוגדרת באופן הבא:

הfonקציה מקבל שם (name) המציין מחרוזת ומחזיר `True` במקרים הבאים:

- עבר או רוע מסוג פגisha: אם האדם ששמו name משתתף בפגisha.

- עבר או רוע מסוג שיחת טלפון: אם האדם אליו מתקשרים הוא name.

בכל יתר המקרים יוחזר `False`.

כתבו את הפונקציה `match` על פי התיאור לעיל בכל המחלקות הרלוונטיות כדי לבצע את הנדרש (רק במחלקות שיש בהן צורך), לפי עקרונות תכונות מונחה עצמים. עברו כל גרסת `match` שאותם כתובים, צינו את שם המחלוקת בה היא כתובה. אם לא מימוש במחלוקת מסויימת את הפונקציה צינו זאת במפורש.

הערה: אין להוסיף תכונה/ות למחלקות.

התשובה היא:

in class Meeting:

def match(name):

for mem in self.\_names:

if mem == name:

return true

return False

in class phonecall:

def match(name):

return self.\_name == name

-2

(5.4)

Event #

in class phonecall:

:def match(self, name)

return False

match



31 DEC 18 31 DEC 18 31 DEC 18

רְאֵבֶן (Eben)

- 10 -



## סעיף ה : (5 נקודות)

כתבו פונקציה בשם `match_events` המתקבלת רשימה של אירועים (אובייקטים מסווג `Event` או מחלקות הירושות ממנה), `events` ושם, `name`. הפונקציה תחזיר רשימת אירועים בה לוקח חלק `name`. במידה ואין אף אירוע מתאים, יש להחזיר `None`.  
יש לכתוב את קוד הפונקציה תוך שימוש בעקרון הפולימורפים.  
הערה : הניתנו כי הפונקציה כתובה באותו קוד במודולות המחלקות.

**התשימת הפונקציה :**

```
def match_events (events, name):
```

```
    res = 0
    for event in events:
        if (5.5) is instance(event, Meetcake) or
           is instance(event, phonecall):
            if event.match(name):
                res += 1
            result.append(event)
        if res == 0:
            return None
        else:
            return result
```

הכל בודק אם הוא מתייחס למשתנה `name` או לא. אם כן, מוסיף אותו לרשימת `result`.  
אם לא, מוסיף אותו לרשימת `result` אם הוא מתייחס למשתנה `name`.  
בסוף, אם הרשימה `result` לא הייתה מזנינה, מeturn `None`.  
בכל מקרה, מeturn `result`.

**שימו לב, כל התשובות צריכות להיבחר על גבי שאלון  
הבחינה. תשובה שתיכתב במקום אחר לא תיבדק!**

**ב ה צ ל ח ה!**

