



דף תרגילים מס' 10

(1) כתבו פונקציה בשם **Q1** המקבלת כפרמטר רשימת מספרים שלמים וחיובים. הפונקציה תחליף כל מספר שלישי ברשימה למספר דו-ספרתי שמורכב מהספרה האחרונה של המספר שקדם לו והספרה הראשונה של המספר שלפניו.

דוגמה:

```
>>> Q1([123, 745, 43, 786, 1900, 346, 7778, 3454, 54, 67])
[123, 745, 73, 786, 1900, 16, 7778, 3454, 38, 67]
```

(2) כתבו פונקציה בשם **Q2** המקבלת כפרמטר רשימת מספרים שלמים וחיובים. הפונקציה תחזיר רשימה חדשה שבה כל מספר ייבנה מחדש על פי התנאים הבאים:

- אם כמות הספרות האי-זוגיות של המספר גדולה מכמות הספרות הזוגיות, המספר ייבנה מהספרות האי-זוגיות בלבד.
- אם כמות הספרות הזוגיות גדולה מכמות הספרות האי-זוגיות, המספר ייבנה מהספרות הזוגיות בלבד.

דוגמה:

```
>>> Q2([132, 6789, 5437, 26, 765, 12586])
[13, 537, 26, 75, 286]
```

(3) כתבו פונקציה בשם **Q3** המקבלת כפרמטרים שתי רשימות של מספרים שלמים וחיובים. הפונקציה תחזיר **True** אם שתי הרשימות מורכבות מאותן ספרות בדיוק, כאשר לכל ספרה אותה כמות הופעות (כולל ספרות שחוזרות). אחרת, תחזיר **False**.

דוגמאות:

```
>>> Q3([258, 10, 86, 0, 755], [6550127, 5088])
True
>>> Q3([258, 1, 86, 75, 54], [655127, 588])
False
```

(4) כתבו פונקציה בשם **Q4** המקבלת כפרמטר רשימת מספרים שלמים וחיובים. הפונקציה תחזיר רשימה חדשה שבה כל המספרים מסודרים בסדר עולה לפי הספרה הגבוהה (לפי מיקום) ביותר בכל מספר.

דוגמה:

```
>>> Q4([854, 8000, 12, 1, 6, 3, 912, 345, 54, 65])
[12, 1, 3, 345, 54, 6, 65, 854, 8000, 912]
```

(5) כתבו פונקציה בשם **Q5** המקבלת כפרמטר רשימת מספרים שלמים וחיובים. הפונקציה תחזיר רשימה חדשה שבה כל המספרים מסודרים לפי כמות הספרות הזוגיות שיש בכל מספר (בסדר עולה).

דוגמה:

```
>>> Q5([124, 29, 331579, 41, 5, 6671, 6280, 81694])
[331579, 5, 29, 41, 124, 6671, 81694, 6280]
```



מבוא לשפת פיתון וקריפטוגרפיה
מיכאל פינקלשטיין

- (6) כתבו פונקציה בשם **Q6** המקבלת כפרמטר רשימת מספרים שלמים וחיוכים ומספר נוסף **n**. הפונקציה תמצא ברשימה את כל המספרים שסכום הספרות שלהם שווה ל-**n**.
- אם אין מספרים שעונים על התנאי, הפונקציה תחזיר 'not found'.
 - אם יש מספר אחד בלבד שעונה על התנאי, הפונקציה תחזיר את המספר עצמו.
 - אם יש כמה מספרים שעונים על התנאי, הפונקציה תחזיר רצף (**tuple**) של כל המספרים האלו.

דוגמאות:

```
>>> Q6( [ 64, 127, 331, 4114, 17111, 6671, 3124 ], 10 )
( 64, 127, 4114, 3124 )
>>> Q6( [ 647, 127, 331, 4164, 17111, 6671, 8124 ], 10 )
127
>>> Q6( [ 647, 627, 331, 4164, 17111, 6671, 8124 ], 10 )
'not found'
```

בהצלחה !!!