



## עבודת הגשה מס' 2

**תאריך הגשה – 29/11/2022**

✓ ניתן להכין את המטלה בזוגות רק חבר אחד בצמד יגיש בפועל את העבודה (במידה ומוגש כעבודה זוגית, יש לרשום בהערה את שמות המגישים ואת מספרי הזהות שלהם). יש להגיש את קבצי הפיתרון תחת שם המכיל את מספרי ת"ז של המגישים.

✓ את החלק התיאורטי יש להגיש בפורמט PDF ואת החלק המעשי יש להגיש בקובץ נפרד בפורמט PY.

✓ חובה להשתמש בשמות וחתימות הפונקציות כפי שמוגדרות בעבודה – אי-עמידה בדרישה הזאת עלולה לגרום לפסילה בבדיקה אוטומטית!

✓ שימו לב, הפלט של דוגמאות ההרצה הוא בהתאם לסביבת הפיתוח Python IDLE ( בהרצה מתוך scriptn ).

✓ חובה לכל פונקציה להוסיף doc strings .

✓ הגשה דרך מודל בלבד!

✓ כל שאלה בנוגע לתרגיל יש להפנות אך ורק לאחראי על התרגיל – גב' סבטלנה

רוסין באימייל: [sceassign2016@gmail.com](mailto:sceassign2016@gmail.com) . פניות בכל בדרך אחרת – לא

יענו! בפנייה, יש לציין את : שם הקורס ופרטים מזהים.

✓ אישורי ההארכה יינתנו ע"י מרצה בלבד !

\* שימו לב: קיים הבדל עקרוני בין הדפסה לבין החזרה של ערך מפונקציה! ברירת המחדל בהיעדר הוראת הדפסה מפורשת היא החזרה בלבד.



## חלק א: שאלות תאורטיות

ענו אם הקביעות הבאות הינן נכונות ונמקו:

- ד. פונקציה מסדר גבוה ( high order function ) זוהי פונקציה מובנת ( built in ) של שפה , שמקבלת פונקציות משתמש כארגומנטים.
- ה. ב- Python 3 משתמשים בהצהרת global על מנת לעדכן קשירה של משתנה במבגרת X מעל מסגרת הנוכחית ( X אינה בהכרח מסגרת גלובלית ).
- ו. אובייקטים מסדר ראשון ( first order object ) ניתן להחזיר מפונקציה ולהעביר לפונקציה כארגומנט.
- ז. לפי מודל הסביבות ( environment model ) הגדרת פונקציה חדשה מוסיפה קשירה חדשה לסביבה גלובלית.

## חלק ב: דיאגרמות סביבה

(5) צייר דיאגרמת סביבה עבור קוד הבא. ציין מה יופיע בהדפסה:

```
x=5
def func1(x,y=2):
    def func2(z):
        global x
        def func3(z):
            nonlocal y
            z+=5
            print('1-{0},{1},{2}'.format(x,y,z))
            y+=z
            return z
        x+=func3(2)
        z=z-1
        print('2-{0},{1},{2}'.format(x,y,z))
        return z
    return func2(x)+2
print(func1(2,3))
print(func1(5))
```



**(6)** צייר דיאגרמת סביבה עבור קוד הבא. ציין מה יופיע בפלט (או הדפסה). (שימו לב שיש מספר הרצות שונות):

```
z = 0
def f(x, y):
    def g(x):
        nonlocal y
        global z
        z = z+1
        y = y-1
        x = x*y
        print("z={0}, y={1}, x={2}".format(z, y, x))
    return g
f = f(5, 10)
f(7)
for n in range(-1,1):
    f(n)
```



(7) פונקציות מסדר גבוה ופונקציות ללא שם (lambda). צייר דיאגרמת סביבה עבור קוד הבא. ציין מה יופיע

בהדפסה:

```
a = 0
def f1(x):
    def f2(y):
        nonlocal x
        x = x * 2
        return x > y
    def f3(y):
        global a
        a = a + 1
        return a + y
    def f4(y):
        nonlocal x
        x = x + 1
        return y + x

    y = 'a'
    if f2(x):
        y = f3(x)
    print('a={0}, x={1}, y={2}'.format(a, x, y))
    return f4

f1(5)(10)
f1(10)
f1 = f1(5)
```



### חלק ג: מעשי: פונקציות מסדר גבוה, פונקציות lambda

(8) כלל סימפסון מתאר שיטת אינטגרציה נומרית אשר משתמשת בפרבולות בין כל שתי נקודות של פונקציה בתור קירוב לפונקציה האמתית. בשיטה זאת, אינטגרל מסוים של פונקציה  $f$  בין  $a$  ל- $b$  מקורב ע"י הנוסחה:

$$\frac{h}{3} [y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n]$$

(9) עבור  $n+1$  נקודות במרווחים שווים בין  $a$  ל- $b$ , כש- $n$  זוגי, הנקודה הראשונה היא  $a$ , והאחרונה היא  $b$ .

$$h=(b-a)/n$$

למספר שלם זוגי  $n$  ו-  $y_k = f(a+kh)$  (כך שכל  $n$  גדול יותר הקירוב יהיה מדויק יותר).  
הגדירו פונקציה **simpson\_rule** שמקבלת ארגומנטים הבאים:  $f, a, b, n$ , ומחזירה ערך של אינטגרל המחושב לפי כלל סימפסון. למשל, על מנת לחשב אינטגרל של  $\text{cube}$  בין 0 ל-1 עם  $n=100$  ניתן להשתמש בפונקציה המוגדרת באופן הבא:

```
<<< simpson_rule(cube,0,1,100)
0.2500000000000000006
```

ראו מידע נוסף ב:

[https://en.wikipedia.org/wiki/Simpson's\\_rule](https://en.wikipedia.org/wiki/Simpson's_rule)

(10) עליך לכתוב משחק. המחשב מגריל מספר תלת-סיפרתי ומאפשר למשתמש לגלות מהו המספר, באמצעות שישה רמזים שהמחשב מעלה בצורה אקראית. בתום כל רמז ניתנת האפשרות להאזין את המספר. כל רמז ואפשרות הקלדת מספר מורידה 10 מ-100 נקודות התחלתיות. במידה והמשתמש גילה את המספר הנכון מציגים לו את כמות הנקודות שהוא זכה במידה שלא נשארו נקודות יש להדפיס הודעה ולסיים את המשחק. עליך לכתוב פונקציה בשם **Game** שמפעילה את המשחק. יש לחבר את הרמזים הבאים בהתאם להנחיות (דוגמאות לפי מספר 186):

- א. סכום כל הספרות. לדוגמא יודפס: "sum: 15". חייבים להגדירה כפונקציה ללא שם (lambda).
- ב. מכפלת כל הספרות. לדוגמא יודפס: "mul: 48". חייבים להגדירה כפונקציה ללא שם (lambda).
- ג. הצגת ספרות זוגיות. לדוגמא יודפס: "even digits: -XX". הכוונה שכל סיפרה זוגית מסומנת ב-"X" וכל סיפרה אי-זוגית ב-".
- ד. הצגת ספרות גדולות. לדוגמא יודפס: "big digits: -XX". הכוונה שכל סיפרה גדולה מ-5 מסומנת ב-"X" וכל סיפרה אחרת ב-".
- ה. הצגה, אם סדר הספרות עולה. לדוגמא יודפס: "ascending: False". הכוונה אם סדר עולה מדפיסים – True, אחרת – False.
- ו. הצגת כל הספרות שהן מספרים ראשוניים (לא כולל 1 ו-0). לדוגמא יודפס: "prime digits: ---". הכוונה שכל סיפרה ראשונית מסומנת ב-"X" וכל סיפרה אחרת ב-".



### הנחיות לתכנות ומבנה הפונקציה:

- אין להשתמש בפונקציות עזר מחוץ ל- **Game()**
- יש פונקציה מסדר גבוה **printF1(msg,f)** שמקבלת כארגומנט מחרוזת (הודעה) ופונקציה, מפעילה את f כדי לבצע הדפסה בהתאם לרמז. יהיה שימוש בפונקציה הזאת רק עבור רמזים א', ב' וה'!
- יש פונקציה מסדר גבוה **printF2(msg,f)** שמקבלת כארגומנט מחרוזת (הודעה) ופונקציה, מפעילה את f (כמה פעמים שצריך) כדי להדפיס את התווים 'X' ו-'-' בהתאם לרמז. יהיה שימוש בפונקציה הזאת רק עבור רמזים ג', ד' וו'!
- יש להיצמד לסגנון הקלט/פלט שמוצג ב-3 דוגמאות הרצה המצורפות.
- התרגיל ייבדק לפי תקינות המשחק ע"פ הכללים שנקבעו וגם מקוריות המימושים. יש יותר מפתרון אחד "נכון".

```
>>> (704)
welcome to game!

ascending order: False
enter number/Enter to finish :452

prime digits: X--
enter number/Enter to finish :656

-----
big digits: X--
enter number/Enter to finish :567

game over !!!
---
```

```
>>> (368)
welcome to game!

sum: 17
enter number/Enter to finish :345

big digits: -XX
enter number/Enter to finish :234

sum: 17
enter number/Enter to finish :567

ascending order: True
enter number/Enter to finish :

game over !!!
>>>
```

```
>>> (186)
welcome to game!

prime digits: ---
enter number/Enter to finish :835

mul: 48
enter number/Enter to finish :345

sum: 15
enter number/Enter to finish :356

ascending order: False
enter number/Enter to finish :243

even digits: -XX
enter number/Enter to finish :456

big digits: -XX
enter number/Enter to finish :186

yes, correct! you win 50 points
>>>
```

**בהצלחה !**