מבוא לתכנות 52304

תרגיל 8 - איטרטורים/גנרטורים ותכנות מסדר שני

שימו לב: בפתרון תרגיל זה אין לעשות שימוש באף מודול חיצוני של **python** – כלומר, אין לעשות import לאף מודול לצורך הפתרון! בפרט, אין לעשות שימוש במודול math או במודול itertools.

שימו לב! בגנרטורים, עליכם להשתמש בפקודה yield.

חלק א: איטרטורים וגנרטורים

חלק זה מורכב ממספר משימות בלתי תלויות. ניתן להניח תקינות הקלט בכל המשימות. על כל הפונקציות עליכם לכלול באותו קובץ ex8_iterators.py.

1. ממשו גנרטור בשם fibonacci המחזיר את איברי סדרת פיבונאצ'י לפי הסדר. סדרה זו מוגדרת באופן הבא:

$a_0=1,\ a_1=1,\ \ a_n=a_{n-1}+a_{n-2}$ for $n>1$

דוגמא לשימוש:

```
>>> fib = fibonacci()
>>> for i in range(10):
    print(next(fib), end=',')
1,1,2,3,5,8,13,21,34,55,
```

2. לצורך סעיף זה בלבד מותר להשתמש בחבילת random שכבר נתקלתם בה בעבר. ממשו גנרטור בשם take_card המחזיר tuple ובו ערך של קלף מקרי מחבילת קלפים סטנדרטית בת 52 קלפים, כאשר הקלפים מוגרלים ללא החזרה (כלומר, לאחר שהקלף נשלף מהרשימה הוא לא מוחזר מהרשימה; למשל, אחרי השליפה הראשונה חבילת הקלפים תכיל 51 קלפים). ה-tuple שיוחזר צריך להיות באורך 2, כאשר באיבר הראשון מופיע ערך הקלף (מספר שלם מ-1 עד 13 כאשר המספר 1 מציין אס והמספרים 11,12,13 מציינים נסיך, מלכה ומלך בהתאמה) ובאיבר השני מופיע הסימן של הקלף, (אחת מארבעת המחרוזות: diamond, heart, club, spade).
דוגמת שימוש:

```
>>> card_iter = take_card()
>>> print(next(card_iter))
(5, 'heart')
>>> print(next(card_iter))
(12, 'diamond')
```

לאחר שנשלפו כל 52 הקלפים הפקודה next(card_iter) צריכה להחזיר הודעת שגיאה.

3. כתבו גנרטור רקורסיבי המחזיר כל פעם את הספרה הבאה בייצוג בינארי של מספר, כאשר הקריאה היא משמאל לימין. שם הפונקציה יהיה recursive_binary המקבל קלט בודד שיהיה מספר שלם. להלן דוגמא להרצת הפונקציה:

```
for i in recursive_binary(10):
...     print(i)
1
0
1
0
```

חלק ב': תכנות מסדר שני

עליכם להגיש את כל שתידרשו לממש בחלק זה בקובץ ex8_2nd_order.py

1. ממשו פונקציה differential_function(g,delta) אשר מחזירה את הפונקציה שהיא הנגזרת הנומרית של הפונקציה g עם חלון ברוחב delta*2:

g'(x) = (g(x+delta) - g(x-delta))/(2*delta)

דוגמא לשימוש:

```
# Squared function
>>> def g(x):
    return x*x
>>> diff = differential_function(g, 0.01)
>>> print("g'(5) = ", diff(5))
g'(5) =  9.999999999999787
```

(התשובה המתמטית המדוייקת היא 10)

2. ממשו פונקציה get_integral_func(x0, x1, num_segments) המחזירה פונקציה integral(g) , אשר מחשבת את האינטגרל המסוים של הפונקציה g בתחום [1x0,x] בעזרת שיטת הטרפז:

1) ראשית נחלק את המקטע ל-num_sements מקטעים שווים
2) עבור כל מקטע (למשל [x0,x0+delta]), נחשב את הערך של הפונקציה g בשני קצות הקטע ונחשב את הערך הממוצע.
3) ממוצע זה כפול רוחב הקטע (delta בדוגמא) יהווה קירוב לאינטגרל של הפונקציה במקטע.
4) סה"כ האינטגרל יחושב על ידי סכימה של הקירובים על פני כל המקטעים וערך זה יוחזר על ידי הפונקציה.

דוגמא לשימוש:

```
>>> def g(x):
    return x*x
>>> integral_func = get_integral_func(-10, 10, 1000)
>>> print("The integral of x^2 in [-10,10] is: ", integral_func(g))
The integral of x^2 in [-10,10] is:  666.6679999999999
```

(התשובה המתמטית המדוייקת היא ....666.66666)

3. ממשו את הפונקציה inverse(f, interval, epsilon) אשר מקבלת פונקציה ממשית ורציפה f ורשימה באורך 2 שערכיה מייצגים אינטרוול. על הפונקציה להחזיר פונקציהg שהיא ההפכית לפונקציה f. כלומר $g(x) = f^{-1}(x)$, כך שאם g(x)=y אז מתקיים f(y)=x. הפונקציה g צריכה לחשב את ההפכי עבור כל מספר ממשי x, אך עליה לחפש אותו רק בתוך הקטע המוגדר ע"י interval. אם לא קיים הפכי (או אולי קיים אך מחוץ לקטע זה) על הפונקציה g להחזיר False. מותר להניח שהפונקציה f היא פונקציה מונוטונית (כלומר עולה או יורדת) בקטע interval (אחרת ייתכן שההפכי אינו מוגדר ביחידות וגם אם קיים קשה יותר למצוא אותו). על הפתרון להיות יעיל הרבה יותר מלעבור על פני כל ערכי ה-interval בקפיצות של epsilon. מומלץ להשתמש בחיפוש בינארי על הקטע.
דוגמא לשימוש:

```
>>> def g(x):
    return x*x
>>> sqrt_func = inverse(g, [0, 1000000], 0.00001)
>>> sqrt_func(16.0)
3.9999977161642164
```

הנחיות להגשה

- עליכם להגיש קובץ ex8.zip הכולל את שני קבצי הפייתון שנדרשתם לכתוב וקובץ README
- 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!