

תרגיל בדיקות יחידה – קורס יסודות תשפ"ג

מחזור קובץ – bank.py. קוד זה מדמה בנק בעל 4 חשבונות.
חשבונות הבנק מיוצגים ע"י מערך המייצג את סכום הכסף בכל חשבון.

0	1	2	3
100\$	0\$	0\$	0\$

חשבון הבנק אותו אתם מנהלים הוא באינדקס 0 ויש ברשותכם 100\$.

בבנק זה ישנן 3 אפשרויות:

1. להפקיד.
2. למשוך.
3. להעביר כסף לחשבון אחר.

בבנק זה לא ניתן להיות במינוס. אם ננסה להעביר/למשוך כסף כאשר אין מספיק בחשבון – הבנק ימנע מאיתנו את הפעולה.

לא ניתן לבצע העברה עצמית ולא ניתן להעביר כסף לחשבון שלא קיים.
עלות פעולה (הפקדה, משיכה, העברה) היא 1\$

עליכם לבנות בדיקות יחידה לפונקציות:

מספר בדיקות (מינימלי)	שם הפונקציה	
3	check_not_negative_or_zero(x)	1
2	convert_to_number(x)	2
2	deposit(amount)	3
3	withdraw(amount)	4
4	transfer(to, amount)	5
2	withdraw_check_balance(amount)	6

*שימו לב – בפונקציה transfer קיים באג. עליכם לכתוב בדיקת יחידה שתגלה את הבאג - הבדיקה צריכה להיכשל.

הערה: בקובץ הבדיקות- הפונקציה setUp היא פונקציה שמורה בunittest. כאשר בדיקה רצה - הערכים של המערך משתנים והבדיקה הבאה תעבוד עם המערך ששונה. כדי להימנע ממצב זה - פונקציה setUp תדאג שערכי המערך יישארו קבועים בכל הבדיקות [100,0,0,0]. ניתן להעתיק את חלק זה.

```
class TestBank(unittest.TestCase):
    def setUp(self):
        bank.bank_accounts_money = [100, 0, 0, 0]
```

הנחיות:

- את המשימה יש לבצע ביחידים.
- יש לכתוב בדיקות יחידה משמעותיות (לדוגמה: בדיקה 1: האם הפונקציה עובדת כראוי?
- בדיקה 2: האם אפשר למשוך יותר ממה שיש בחשבון?).
- בדיקה שתחזור על עצמה לא תחשב (לדוגמה: בדיקת הפקדה של 30\$ ובדיקת הפקדה של 20\$ זו בדיקה אחת).
- יש לצרף לקובץ תיעוד הנעשה ע"י מתן הערות בקוד.
- אופן ההגשה: יש להגיש את קובץ הבדיקות בלבד כאשר שם הקובץ יהיה: **bank_test.py**
- ניתן להוסיף בדיקות יחידה מעל הנדרש.

עבודה מהנה!