

תרגיל 3 – סודוקו (פונקציות, רשימות דו מימדיות)

תאריך פרסום: 05/11/2018

תאריך הגשה: 12/11/2018 בשעה 23:59

מתרגל אחראי: שגיא תובל

הנחיות כלליות:

- קראו את כל ההוראות לגבי הגשת תרגילי הבית באתר הקורס.
- קראו את כל העבודה לפני שתתחילו לפתור אותה.
- כתבו תיעוד (הערות) שמסביר את הקוד שלכם. אין לכתוב הערות בעברית.
- שאלות בנוגע לעבודה יישאלו ב-"פורום שאלות לתרגיל בית 3" במודל או בשעות הקבלה של המתרגל האחראי בלבד.
- כתבו את כל הקוד שלכם בקובץ 'hw3.py'.
- השימוש בחבילות חיצוניות אסור בהחלט ויגרור ציון 0.
- יש להקפיד על חתימות זהות של הפונקציות, שכן מערכת הבדיקה קוראת לפונקציות בצורה אוטומטית.

בעבודה זו נכתוב תוכנית המממשת את המשחק "סודוקו".

את לוח המשחק נייצג ע"י רשימה של רשימות (רשימה דו מימדית) המייצגות את השורות. כל רשימה המייצגת שורה מכילה את הספרות בעמודות השונות. תאים שעדיין לא מולאו ייוצגו ע"י הספרה 0. ניתן להניח שלוח המשחק כבר נתון.

להלן דוגמא ללוח סודוקו וייצוגו ברשימה דו מימדית:

```
board = [[3,7,0,0,5,0,0,0,0],
          [0,6,0,0,3,0,2,0,0],
          [0,2,9,4,0,0,0,7,8],
          [0,0,4,1,7,0,0,8,0],
          [0,0,6,3,0,5,9,0,0],
          [0,5,0,0,8,4,1,0,0],
          [7,1,0,0,0,8,5,6,0],
          [0,0,5,0,1,0,0,2,0],
          [0,0,0,0,9,0,0,1,3]]
```

| | | | | | | | | |
|---|---|---|---|------|---|---|---|-------|
| 3 | 7 | | | 5 | | | | |
| | 6 | | | 3 | | 2 | | |
| | 2 | 9 | 4 | שורה | | | 7 | 8 |
| | | 4 | 1 | 7 | | | 8 | |
| | | 6 | 3 | | 5 | 9 | | עמודה |
| | 5 | | | 8 | 4 | 1 | | |
| 7 | 1 | | | | 8 | 5 | 6 | |
| | | | 5 | | 1 | | 2 | |
| | | | | 9 | | | 1 | 3 |

חלק א' – בדיקת תקינות לוח ומהלך

ממשו את הפונקציה `is_valid_sudoku_board(board)` המקבלת לוח (`board`) ובודקת האם הוא תקין.

הבדיקות הנדרשות לוודא תקינות לוח הן:

1. ה-`type` של הלוח ושל כל שורה בו.
2. מספר השורות בלוח.
3. מספר האיברים בכל שורה.
4. ה-`type` והערך של כל ספרה בכל שורה.
5. מספר העמודות בלוח.

במידה וכל הבדיקות יעברו בהצלחה הפונקציה תחזיר `True`, אחרת `False`.

תזכורת: הפונקציה השמורה `type` מקבלת איבר ומחזירה את סוגו. לדוגמא – הקריאה `type('Hello')` תחזיר `str`.

ממשו את הפונקציה `is_valid_move(board, row, col, digit)`. הפונקציה מקבלת לוח משחק (`board`), שורה (`row`) מסוג `int`, עמודה (`col`) מסוג `int` וספרה (`digit`) מסוג `int`. הפונקציה מבצעת את הבדיקות הבאות לאישור המהלך:

1. הלוח הינו לוח סודוקו תקין.
2. הערך של `row`, `col` ו-`digit` בטווח הנדרש (`digit` חייב להיות גדול מ-0, כלומר לא ניתן לאפס תא עם ערך).
3. התא אליו מבקשים להכניס את הספרה ריק.
4. השורה הנתונה בלוח המשחק לא מכילה את הספרה.
5. העמודה הנתונה בלוח המשחק לא מכילה את הספרה.
6. הבלוק בו נמצא התא המבוקש לא מכיל את הספרה.

במידה וכל הבדיקות יעברו בהצלחה הפונקציה תחזיר `True`, אחרת `False`.

ניתן להניח כי הסוג של `row`, `col` ו-`digit` הוא `int`.

כמו כן, כדי להכניס לשורה \ לעמודה הראשונה יינתן האינדקס 0 ב-`row` ו-`col` בהתאמה.

רמז: נשים לב שהבלוק בוא נמצא התא המבוקש מתחיל בשורה ה- $3 * \text{int}(\frac{row}{3})$ ובעמודה ה- $3 * \text{int}(\frac{col}{3})$.

מה יקרה אם לא נבצע `casting` ל-`int`?

חלק ב' – הכנסת איבר ללוח

ממשו את הפונקציה `enter_digit(board, row, col, digit)`. הפונקציה מקבלת לוח משחק (`board`), שורה (`row`) מסוג `int`, עמודה (`col`) מסוג `int` וספרה (`digit`) מסוג `int`. במידה והמהלך חוקי, הפונקציה

מכניסה את הספרה `digit` לתא המבוקש ומחזירה `True`. אחרת, יוחזר `False`.

ניתן להניח כי כל קלטי הפונקציה תקינים.

הנחיות הגשה:

- יש להגיש את העבודה למערכת ההגשה כפי שמתואר בהנחיות ההגשה במודל.
- יש להגיש קובץ מכוון אחד בלבד בשם 'hw3.zip' (לא rar או שום סיומת אחרת), כאשר בתוכו נמצאת תיקייה בשם hw3 ובתוכה הקובץ 'hw3.py'. כל מבנה, שם אחר או סיומת אחרת ייגררו ציון 0.

😊 בהצלחה!