



שאלה 5 (20 נקודות)

משחק סודוקו מורכב מלוח בגודל 9×9 של ספרות בטווח 1-9, ממולא חלקית. לדוגמה:

			1			7	4	
	5			9			3	2
		6	7			9		
4			8					
	2						1	
					9			5
		4			7	3		
7	3			2			6	
	6	5			4			

מטרת השחקן למלא את יתרת המשבצות בלוח בספרות 1-9 כך ש:

- באותה שורה לא תופיע אף ספרה יותר מפעם אחת.
- באותו טור לא תופיע אף ספרה יותר מפעם אחת.

שימו לב שבמשחק המלא יש תנאי נוסף, אך אנו לא מתייחסים אליו בשאלה זו.

לדוגמה, הנה פתרון ללוח למעלה:

3	9	2	1	8	5	7	4	6
8	5	7	4	9	6	1	3	2
1	4	6	7	3	2	9	5	8
4	7	9	8	5	1	6	2	3
5	2	8	6	7	3	4	1	9
6	1	3	2	4	9	8	7	5
2	8	4	5	6	7	3	9	1
7	3	1	9	2	8	5	6	4
9	6	5	3	1	4	2	8	7

בשאלה זו נייצג לוח סודוקו באמצעות מערך דו-ממדי `board[9][9]`, המכיל מספרים בטווח 1-9 במקומות המלאים, ואפסים במקומות הריקים. עליכם לממש את הפונקציה `() sudoku` (בדף הבא) המקבלת לוח סודוקו מלא חלקית, וממלאת אותו לפתרון מלא כלשהו על ידי החלפת האפסים שבמערך במספרים בין 1 ל-9. ניתן להניח כי הלוח שהפונקציה `() sudoku` מקבלת הינו פתיר.

שימו לב: על הפתרון להיות רקורסיבי ולעבוד בשיטת ה-`backtracking`; בפרט, אין להמשיך ברקורסיה עם לוח שברור כי אינו פתיר. זכרו שביכולתכם לממש פונקציות עזר נוספות על פי הצורך.



```
void sudoku(int board[9][9]) {
    sudoku_aux(board);
}

int sudoku_aux(int board[9][9])
{
    int i,j,num;
    for (i=0; i<9; ++i) {
        for (j=0; j<9; ++j) {
            if (board[i][j]==0) {
                for (num=1; num <= 9; ++num) {
                    if (legal(board,i,j,num)) {
                        board[i][j] = num;
                        if (sudoku_aux(board))
                            return 1;
                        board[i][j] = 0;
                    }
                }
                return 0;
            }
        }
    }
    return 1;
}

int legal(int board[9][9], int r, int c, int num)
{
    int k;
    for (k=0; k<9; ++k) {
        if (board[k][c] == num) return 0;
        if (board[r][k] == num) return 0;
    }
    return 1;
}
```