

# Gra sudoku w python

## Generowanie planszy:

Plansza do gry jest przechowywana jako tablica dwuwymiarowa.

Najpierw generowane są pola na przekątnej, gwarantując brak powtórzeń pionowo i poziomo.

```
1 usage
def fill_diagonal_box(board, row, col):
    """Wypełnianie przekątnych"""
    nums = random.sample(range(1, 10), k=9)
    for i in range(3):
        for j in range(3):
            board[row + i][col + j] = nums[i * 3 + j]
```

5	1	9							1
4	7	6							2
8	2	3							3
			9	5	6				4
			2	3	1				5
			7	8	4				6
						1	2	6	7
						7	4	5	8
						9	8	3	9
a	b	c	d	e	f	g	h	i	

Następnie reszta planszy jest wypełniana.

```
2 usages
def solve(board):
    """Wypełnianie reszty pól"""
    find = find_empty(board)
    if not find:
        return True
    else:
        row, col = find

        for i in range(1, 10):
            if valid(board, i, pos: (row, col)):
                board[row][col] = i

                if solve(board):
                    return True

            board[row][col] = 0

        return False
```

9	8	5	1	2	3	4	7	6	1
6	2	7	4	8	5	1	3	9	2
3	4	1	6	7	9	2	5	8	3
1	5	2	8	6	4	7	9	3	4
8	6	3	7	9	2	5	1	4	5
4	7	9	3	5	1	8	6	2	6
2	9	6	5	1	8	3	4	7	7
5	3	8	9	4	7	6	2	1	8
7	1	4	2	3	6	9	8	5	9
a	b	c	d	e	f	g	h	i	

W sposób losowy usuwane są kolejne elementy z tablicy, tworząc w ten sposób zagadkę. Puste pola są reprezentowane jako '0' w tablicy.

```
def remove_elements(board):  
    """Usuwanie elementów tablicy"""  
    print("Witaj w sudoku, wybierz poziom trudności: \n1. Łatwy\n2. Średni\n3. Trudny")  
    difficulty = input()  
  
    match difficulty:  
        case "1": num_holes = 15  
        case "2": num_holes = 25  
        case "3": num_holes = 35  
        case _: num_holes = 5  
    count = num_holes  
    while count > 0:  
        row = random.randint(a: 0, b: 8)  
        col = random.randint(a: 0, b: 8)  
        while board[row][col] == 0:  
            row = random.randint(a: 0, b: 8)  
            col = random.randint(a: 0, b: 8)  
        board[row][col] = 0  
        count -= 1
```

			6	1			9		1
4	9	1			7			8	2
			8	4	9	2		1	3
1			3	6			8	9	4
	3	8	1	7		5	6	4	5
6	7	5			8	1	2		6
	1	6			4	8			7
5		9			1		4	6	8
7		3	5		6	9		2	9
a	b	c	d	e	f	g	h	i	

## Instrukcja dla użytkownika:

Na starcie programu jesteśmy zapytani o wybranie poziomu trudności. Należy wpisać odpowiednią cyfrę i wcisnąć 'ENTER', np. dla poziomu łatwego jest to '1'.

```
Witaj w sudoku, wybierz poziom trudności:
1. Łatwy
2. Średni
3. Trudny
1|
```

Ruch użytkownika jest wykonywany przez podanie odpowiednich współrzędnych pola w formacie 'kolumna rząd wprowadzana\_cyfra', gdzie kolumny to litery pod planszą (a-i), rzędy to cyfry wypisane po prawej stronie planszy (1-9) i wprowadzana\_cyfra to cyfra, którą chcemy wpisać w dane pole.

6	8	5		1	3	4	7	9	1
4	1	9	5	8	7	3	2	6	2
2	7	3	4	9	6	5			3
	2		1		8	9			4
1	9	7		5	4	8	6		5
5		8	7	2	9		4	3	6
9		6	8	7	5	2	1	4	7
8	4	1	6	3		7	9	5	8
7	5	2	9	4		6	3	8	9
a	b	c	d	e	f	g	h	i	

Np. Chcąc wypełnić pole zaznaczone kolorem czerwonym cyfrą 2, użytkownik musi wpisać polecenie: 'd 1 2'. Jeśli użytkownik spróbuje wykonać nielegalny ruch zostanie wyświetlona odpowiednia informacja.

Gra toczy się, dopóki któreś z pól ciągle pozostaje puste. Można też wprowadzić 'q', aby zakończyć działanie programu.