Backtracking Algorithm

In backtracking algorithms you try to build a solution one step at a time. If at some step it becomes clear that the current path that you are on cannot lead to a solution you go back to the previous step (backtrack) and choose a different path. Briefly, once you exhaust all your options at a certain step you go back.

Approach For Solving Sudoku Using Recursive Backtracking Algorithm

1. Like all other Backtracking problems, we can solve Sudoku by one by one assigning numbers to empty cells.
2. Before assigning a number, we need to confirm that the same number is not present in current row, current column and current 3X3 subgrid.
3. If number is not present in respective row, column or subgrid, we can assign the number, and recursively check if this assignment leads to a solution or not. If the assignment does not lead to a solution, then we try next number for current empty cell. And if none of number (1 to 9) lead to solution, we return false.

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

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

Input

```
Input
9
5 3 0 0 7 0 0 0 0
6 0 0 1 9 5 0 0 0
0 9 8 0 0 0 0 6 0
8 0 0 0 6 0 0 0 3
4 0 0 8 0 3 0 0 1
7 0 0 0 2 0 0 0 6
0 6 0 0 0 0 2 8 0
0 0 0 4 1 9 0 0 5
0 0 0 0 8 0 0 7 9
```

Output

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