

# Solving constraint satisfaction problems using RAT

24 February 2022



# The Team

**Sarthak S Kumar -PES2UG21CS432**

**Varun C -PES2UG21CS593**

**Vishal M Godi -PES2UG21607**

**Section: K**

**Batch: 2**

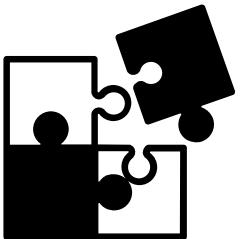


# Constraint satisfaction problems?





It is the process of finding a solution to a set of constraints, that impose conditions that the variables must satisfy. A solution is therefore a set of values for the variables that satisfies all constraints.



**Constraint satisfaction problems** on finite domains are typically solved using a form of search. The most used techniques are variants of backtracking using **recursion**, constraint propagation, and local search.



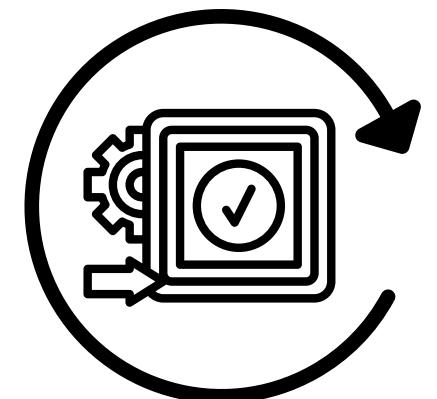
# Recursive algorithms





A recursive function is one that calls itself with simpler values.

In recursion, we try to solve a problem by solving a smaller instance of the same problem, unless the problem is so small that we can just solve it directly.



**Backtracking** is a recursive tool for solving constraint satisfaction problems



# Backtracking



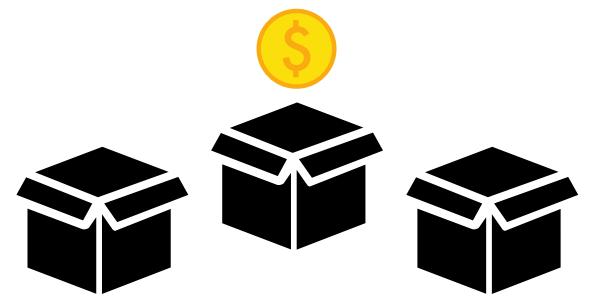
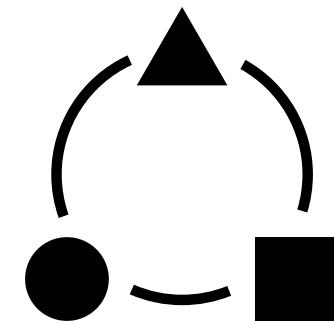


It's an **algorithmic-technique** for solving problems recursively by trying to build a solution incrementally, one at a time removing those solutions that fail to satisfy the constraints of the problem at any point of time

In short words, it's searching **every possible combination** in order to solve a computational problem.

In **recursion**, the function calls itself until it reaches a base case.

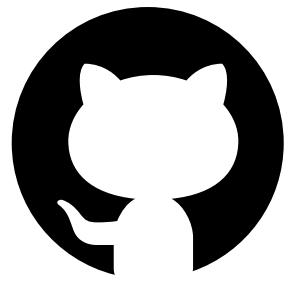
In **backtracking**, we use recursion to explore all the possibilities until we get the best result for the problem.



# Code



**Frameworks:**  
Tkinter



**Link to the project files**

**Modules:**  
Time  
Math  
Random  
Datetime  
PIL  
Numpy  
Copy  
Ctypes



# Project Snapshots



# Solving Constraint Satisfaction Problems using RAT

Varun C  
Vishal M Godi  
Sarthak S Kumar  
Sem 1 CSE, PESU ECC

## Welcome

Select one to continue



Solve Sudoku



Rat in a Maze

Exit

**main.py**





Solve Sudoku

Sudoku  $\frac{1}{2}$

Welcome

Your Name: PESU

Difficulty: Intermediate

Submit

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

✗ You gave up so quick!

Better luck next time

With Experience comes Expertise!

Next

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

No relief mate!

Check it again!

Chances Remaining: 9

Check Solution

Solve Sudoku

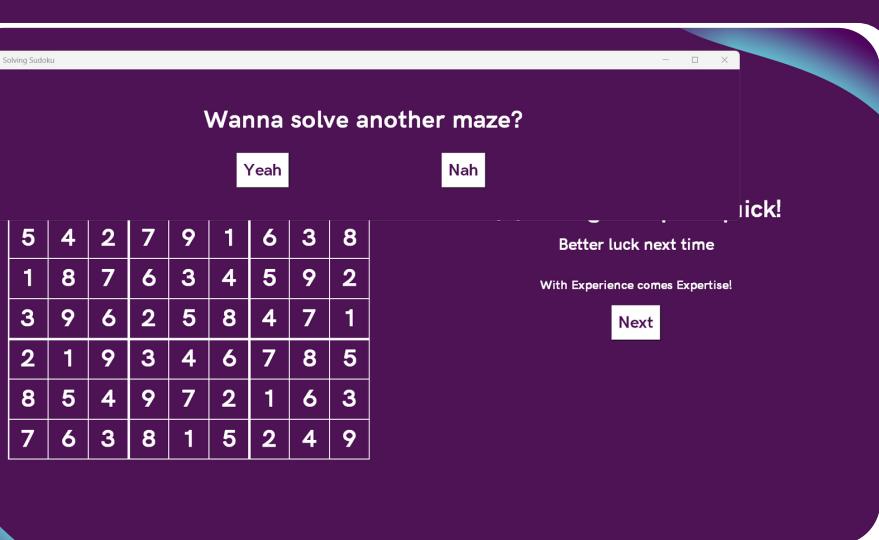
	5		6	3	8	2		
	2	1		9	3	5		
	8		2			6		
5	4			1	6	8		
1		6		4	5	9	2	
3	9			8	4	7	1	
	9		6		8			
5	4	9	7		1	3		
7	6		1	5	2	4	9	

Hello PESU  
Your sudoku is here. Go ahead, Solve it

Use to enter numbers  
and to move the selector

Check Solution

Solve Sudoku

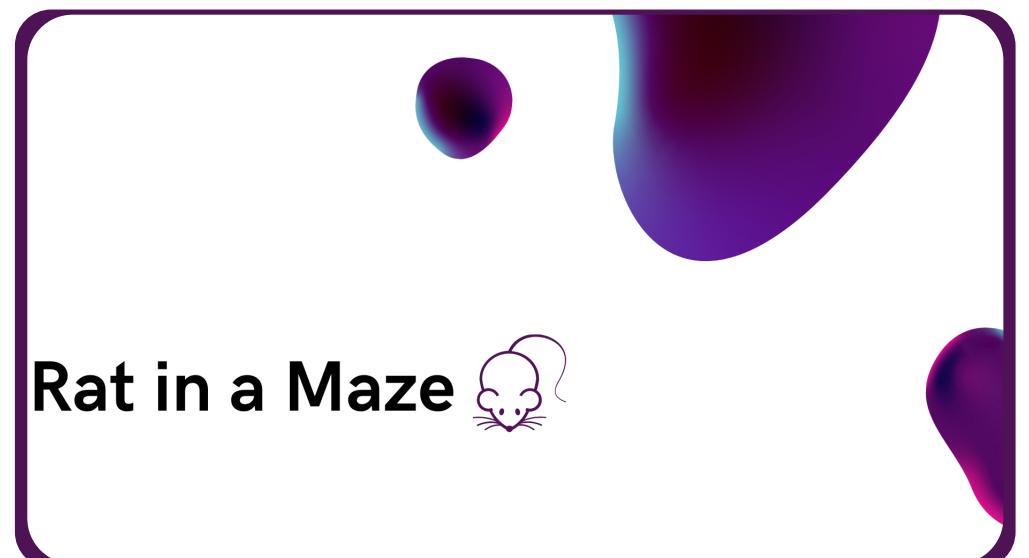


solve\_sudoku\_UI.py

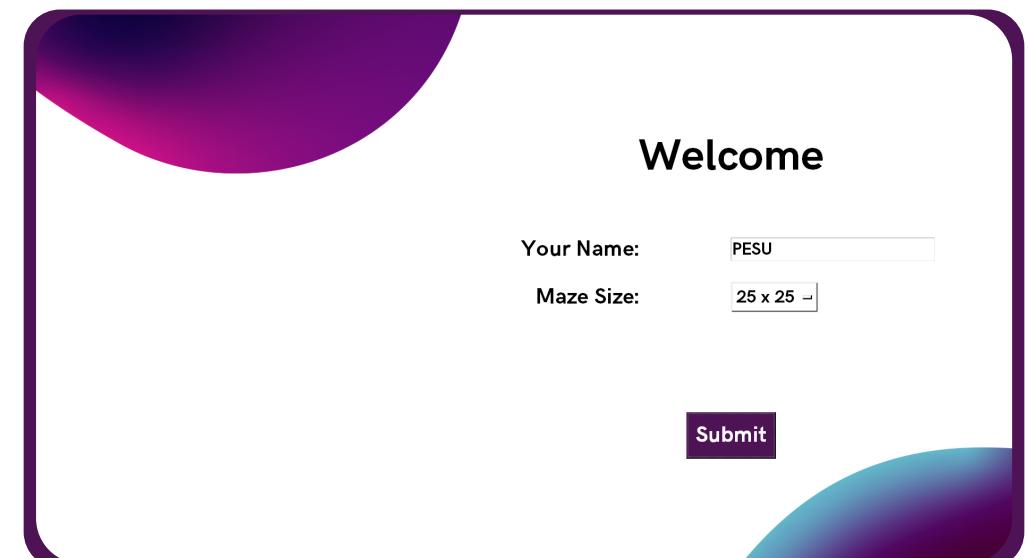




Rat in a Maze



Rat in a Maze

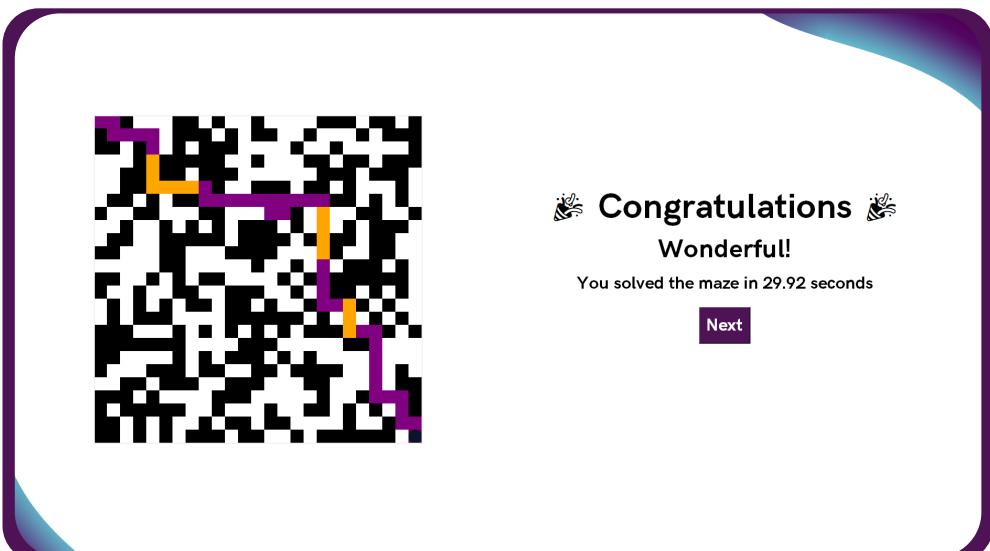


Welcome

Your Name: PESU

Maze Size: 25 x 25

Submit

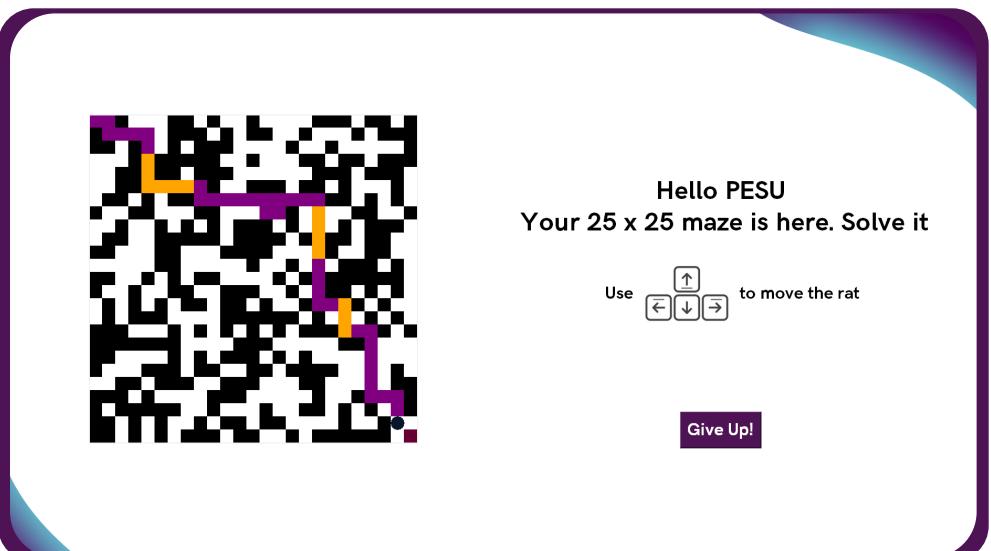


Congratulations!

Wonderful!

You solved the maze in 29.92 seconds

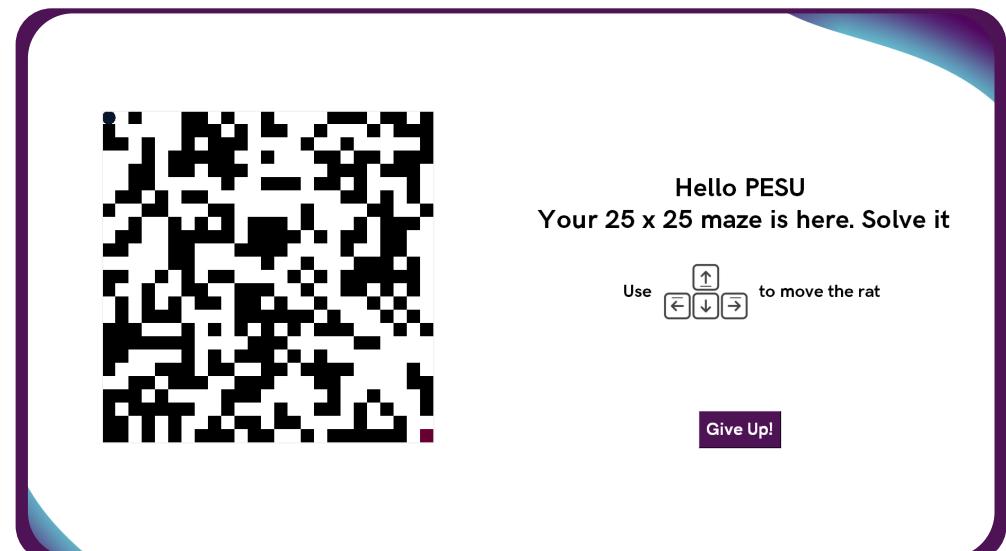
Next



Hello PESU  
Your 25 x 25 maze is here. Solve it

Use to move the rat

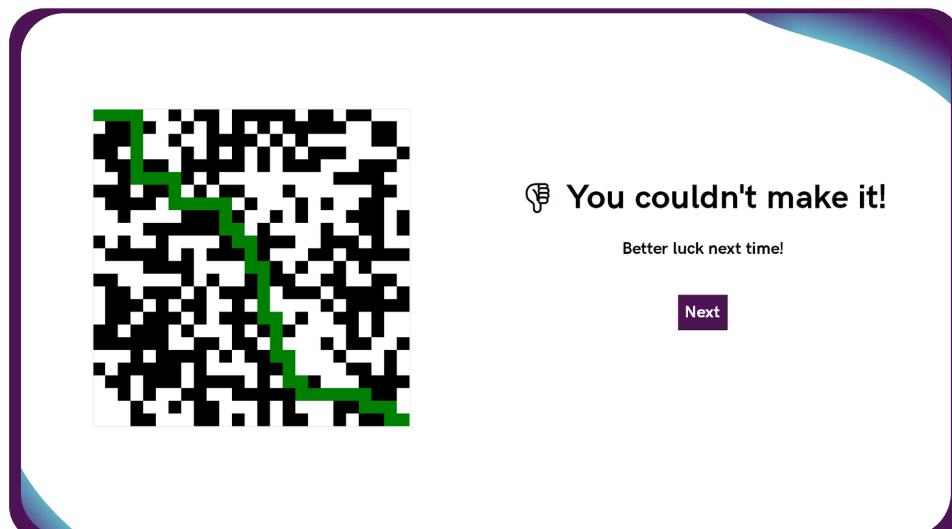
Give Up!



Hello PESU  
Your 25 x 25 maze is here. Solve it

Use to move the rat

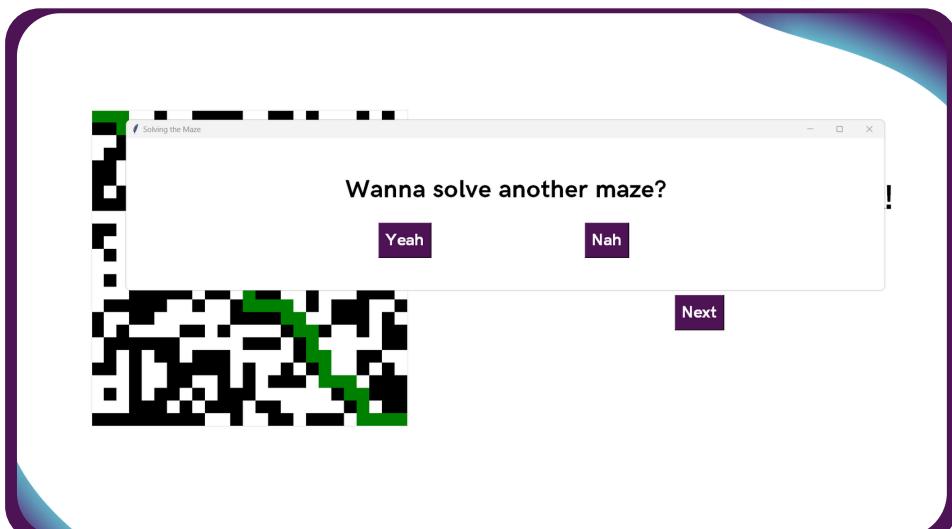
Give Up!



You couldn't make it!

Better luck next time!

Next



Wanna solve another maze?

Yeah

Nah

Next

Rat\_in\_a\_maze\_UI.py



# Thank you!

Solving constraint satisfaction problems using RAT

