

# Solving constraint satisfaction problems using RAT

29 January 2022



# The Team

**Sarthak S Kumar -PES2UG21CS432**

**Varun C -PES2UG21CS593**

**Vishal M Godi -PES2UG21607**

**Section: K**

**Batch: 2**

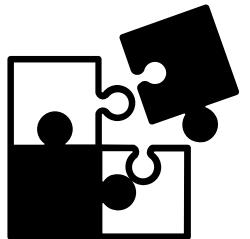


# Constraint satisfaction problems?





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.



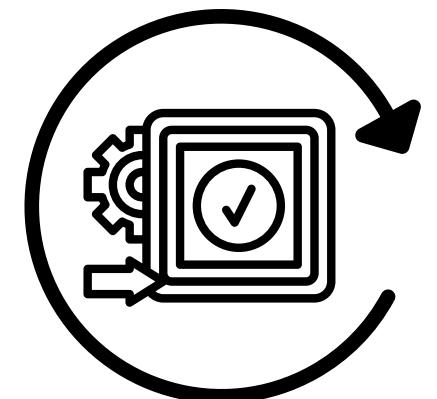
# Recursive algorithms





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



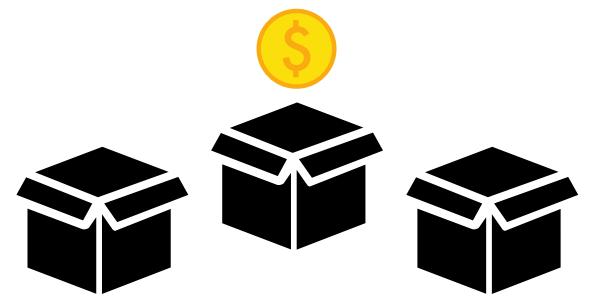
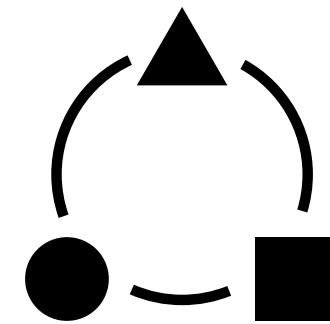


It's an **algorithmic-technique** for solving problems recursively by trying to build a solution incrementally, one at a time removing those solutions that fail to satisfy the constraints of the problem at any point of time

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.

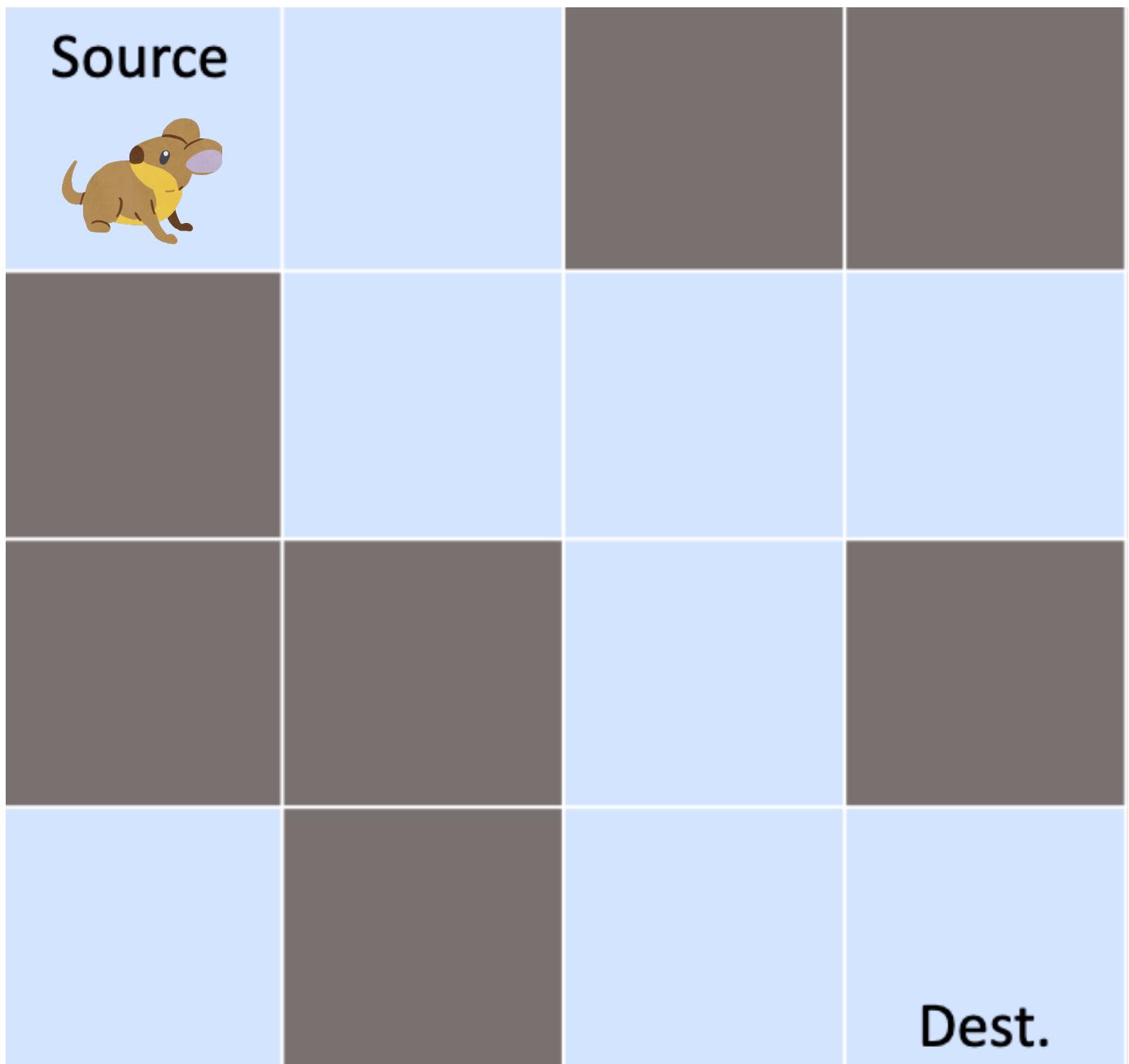
In **backtracking**, we use recursion to explore all the possibilities until we get the best result for the problem.



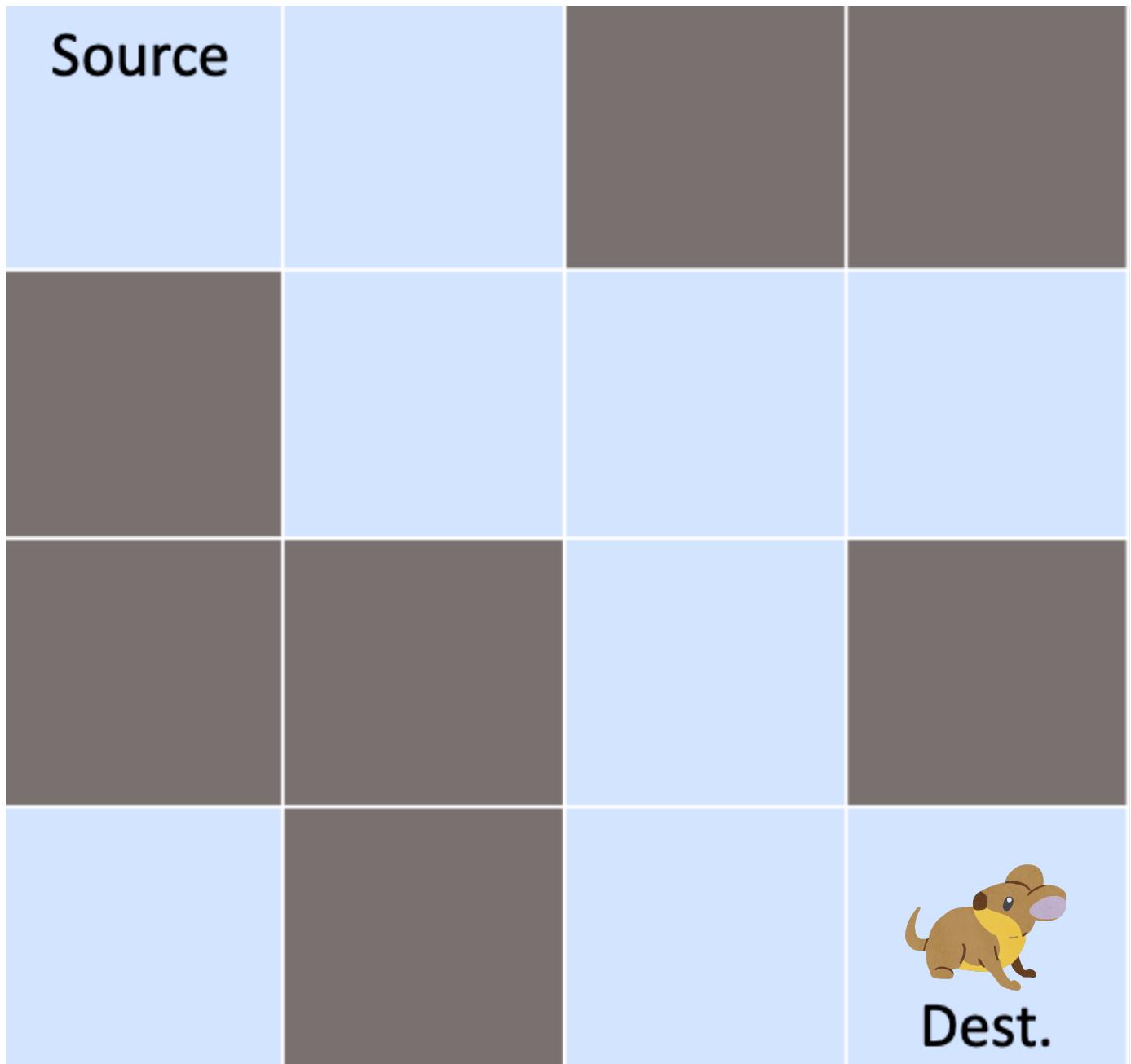
# Implementation



# Rat in a maze



# Rat in a maze



# Sudoku



## Constraints

- Each row must have unique numbers from 1-9
- Each column has unique numbers from 1-9
- Each sub-grid has unique numbers from 1-9

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



# Sudoku

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

The backtracking algorithm here will try to place each number in each row and column following the constraints until the puzzle is solved.



# Sudoku

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

No solution?

Backtrack

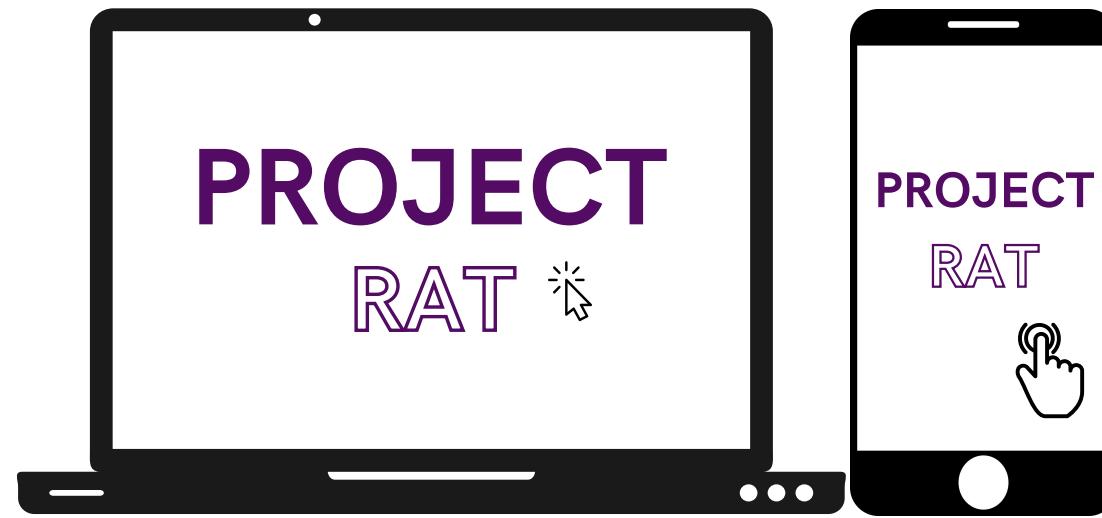
Retry

Found Solution!

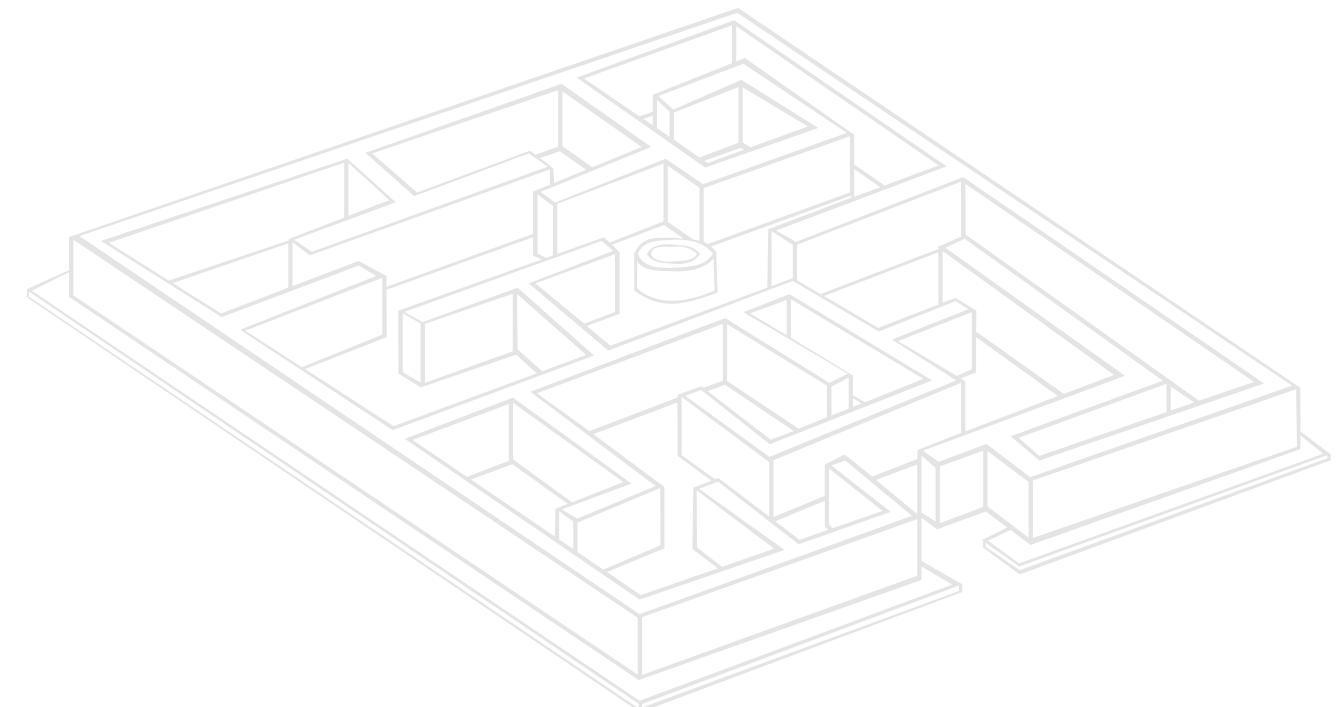
The backtracking algorithm here will try to place each number in each row and column following the constraints until the puzzle is solved.



# UI



Developing a basic UI to facilitate interactions with the reviewers and enable us to present the problems and their solutions in a more organized and interactive manner.

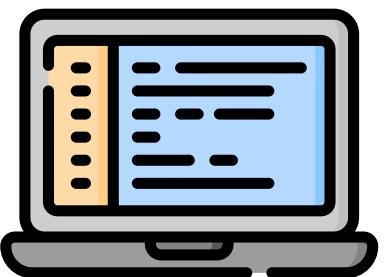
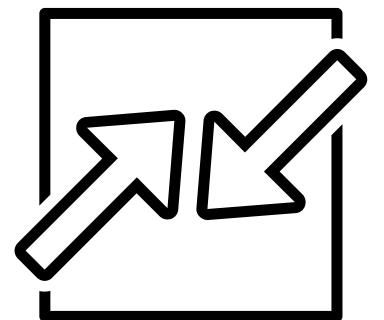
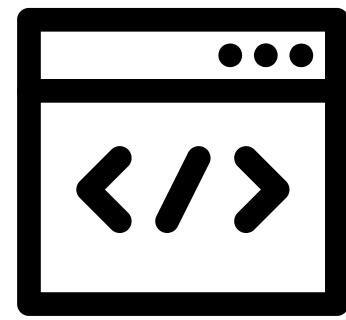
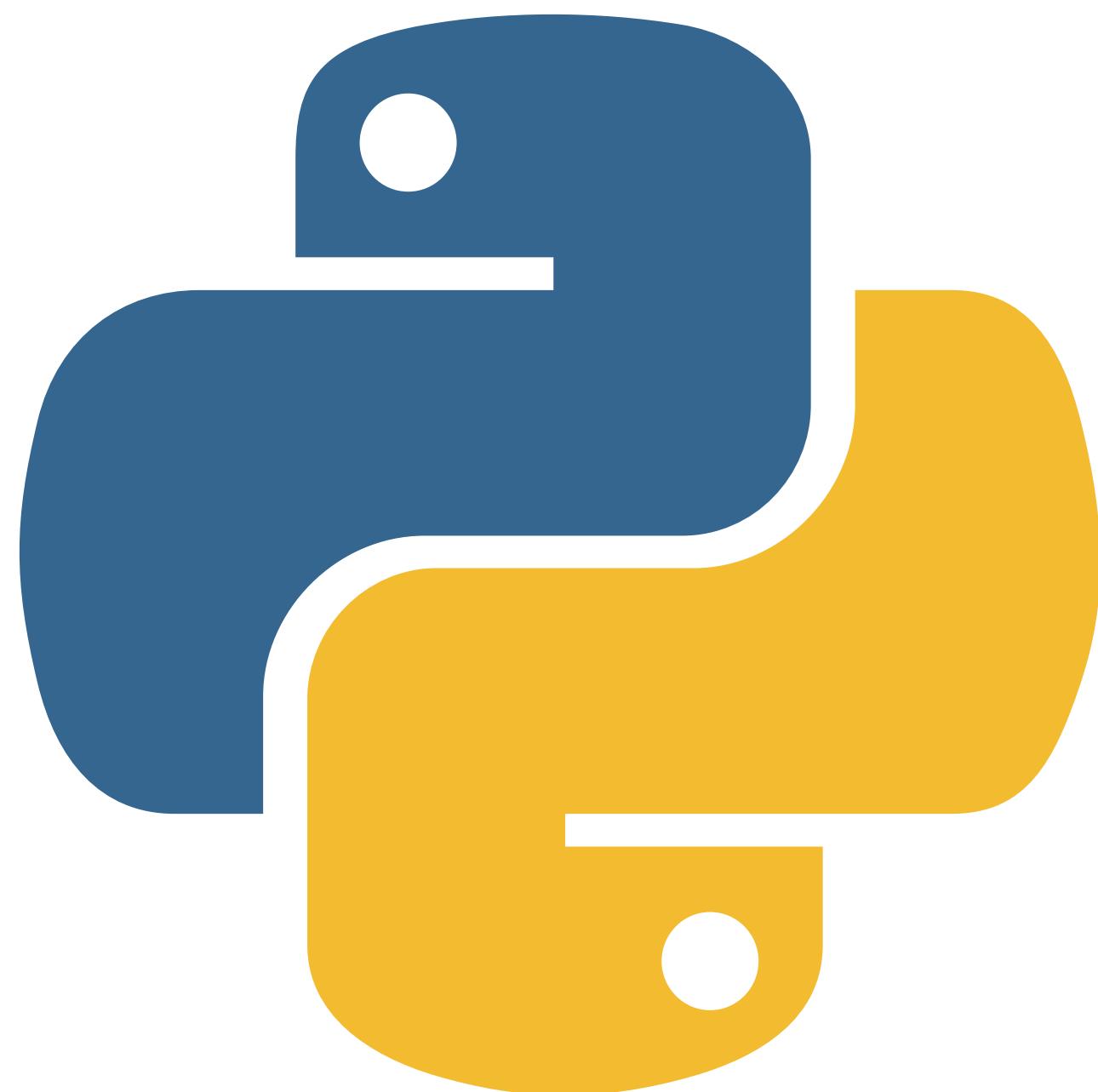


$\frac{1}{2}$



# Technologies and Resources





# Thank you!

Solving constraint satisfaction problems using RAT

