Unit-2

Problem Solving, Problems, Problem Space & Search

Introduction to Problem and its solution

- To build a system to solve a particular problem, we need to do four things:

1. Define the problem precisely.
2. Analyze the problem.
3. Isolate and represent the task knowledge which is necessary to solve the problem.
4. Choose the best problem-solving technique and apply it to the particular problem.

Examples:

- Chess :-

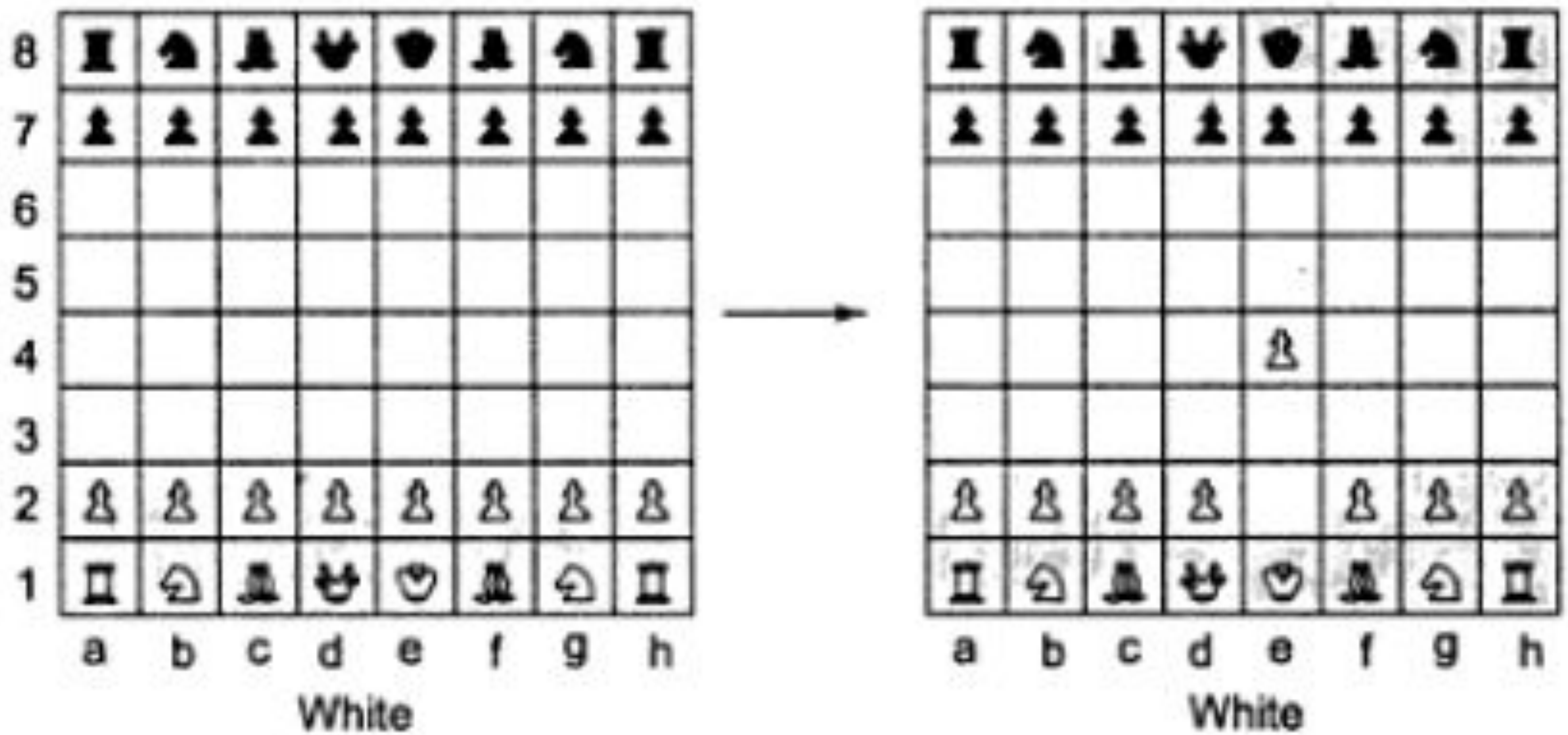


Fig. 2.1 *One Legal Chess Move*

- For the problem “Play Chess”, it is fairly easy to provide a formal and complete problem description.
- 8 x 8 chessboard, allowing only legal moves and can’t be back.
- Around 10^{120} possible board positions
  - No person could ever supply a complete set of such rules.
  - No program could easily handle all those rules.

- The state space representation forms the basis of most of the AI methods and its structure of problem solving in two important ways:
  - It allows for a formal definition of a problem as the need to convert some given situation into some desired situation using a set of permissible operations.
  - It permits us to define the process of solving a particular problem as a combination of known techniques and search.

- Water Jug Problem:-

"You are given two jugs, a 4-gallon one and a 3-gallon one. Neither has any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 gallons of water into the 4-gallon jug?"

4-gallon water jug

3-gallon water jug

1	(x, y) if $x < 4$	$\rightarrow (4, y)$	Fill the 4-gallon jug
2	(x, y) if $y < 3$	$\rightarrow (x, 3)$	Fill the 3-gallon jug
3	(x, y) if $x > 0$	$\rightarrow (x - d, y)$	Pour some water out of the 4-gallon jug
4	(x, y) if $y > 0$	$\rightarrow (x, y - d)$	Pour some water out of the 3-gallon jug
5	$(x, y]$ if $x > 0$	$\rightarrow (0, y)$	Empty the 4-gallon jug on the ground
6	(x, y) if $y > 0$	$\rightarrow (x, 0)$	Empty the 3-gallon jug on the ground
7	(x, y) if $x + y \geq 4$ and $y > 0$	$\rightarrow (4, y - (4 - x))$	Pour water from the 3-gallon jug into the 4-gallon jug until the 4-gallon jug is full
8	(x, y) if $x + y \geq 3$ and $x > 0$	$\rightarrow (x - (3 - y), 3)$	Pour water from the 4-gallon jug into the 3-gallon jug until the 3-gallon jug is full
9	(x, y) if $x + y \leq 4$ and $y > 0$	$\rightarrow (x + y, 0)$	Pour all the water from the 3-gallon jug into the 4-gallon jug
10	(x, y) if $x + y \leq 3$ and $x > 0$	$\rightarrow (0, x + y)$	Pour all the water from the 4-gallon jug into the 3-gallon jug
11	$(0, 2)$	$\rightarrow (2, 0)$	Pour the 2 gallons from the 3-gallon jug into the 4-gallon jug
12	$(2, y)$	$\rightarrow (0, y)$	Empty the 2 gallons in the 4-gallon jug on the ground

Fig. 2.3 *Production Rules for the Water Jug Problem*

Gallons in the 4-Gallon Jug	Gallons in the 3-Gallon Jug	Rule Applied
0	0	
		2
0	3	
		9
3	0	
		2
3	3	
		7
4	2	
		5 or 12
0	2	
		9 or 11
2	0	

Fig. 2.4 *One Solution to the Water Jug Problem*

Production Systems

- A production system consists of :
  - **A set of rules**, each consisting of a left side (a pattern) which determines the applicability of the rule and a right side that describes the operation to be performed if the rule is applied.
  - One or more **knowledge/databases** that contain whatever information is appropriate for the particular task.
  - A **control strategy** that specifies the order in which the rules will be compared to the database and a way of resolving the conflicts that arise when several rules match at once.
  - A **rule applier**.

Problem Characteristics

- In order to choose the most appropriate method or combination of methods for a particular problem, it is necessary to analyze the problem along several key dimensions:

Is the problem decomposable into a set of (nearly) independent smaller or easier subproblems?

Can solution steps be ignored or at least undone if they prove unwise?

Is the problem's universe predictable?

Is a good solution to the problem obvious without comparison to all other possible solutions?

Is the desired solution a state of the world or a path to a state?

Is a large amount of knowledge absolutely required to solve the problem or is knowledge important only to constrain the search?

Can a computer that is simply given the problem return the solution, or will the solution of the problem require interaction between the computer and a person?

1

Is the problem decomposable into a set of (nearly) independent smaller or easier subproblems?

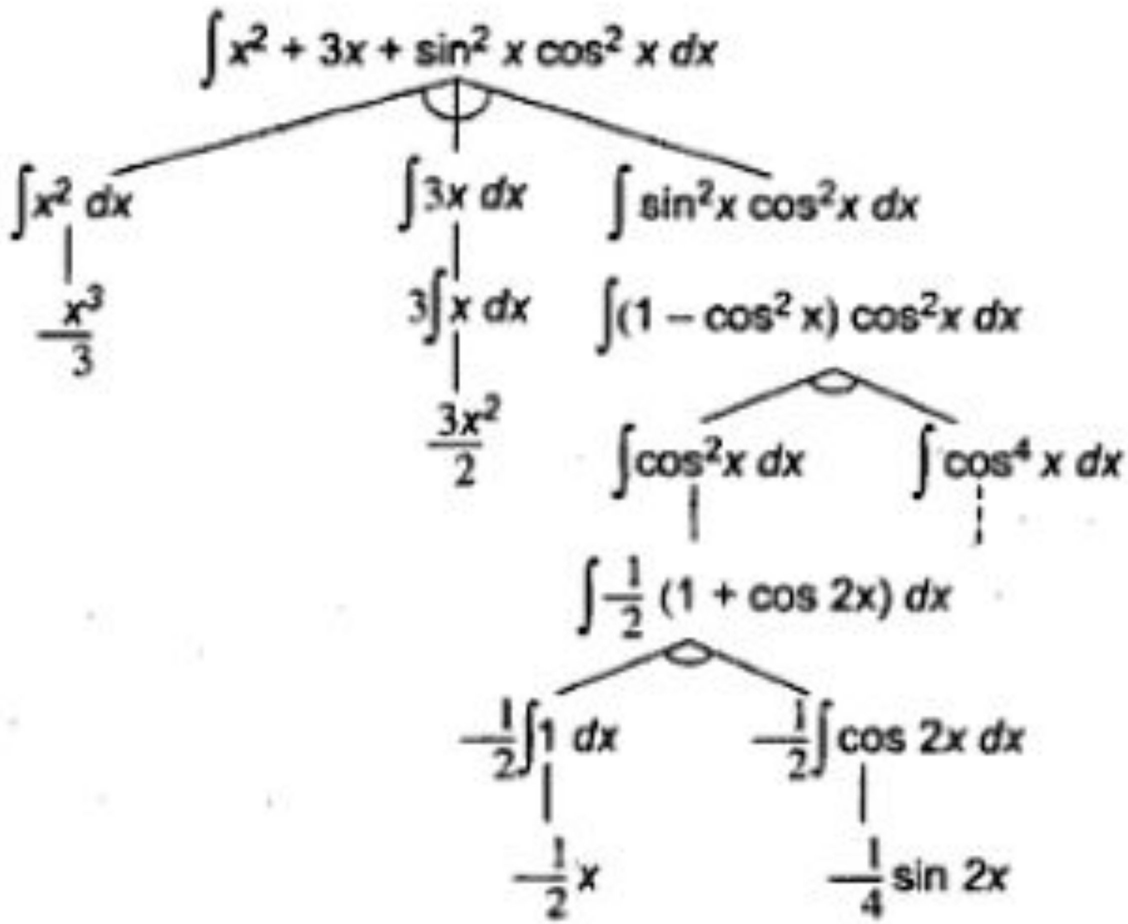


Fig. 2.9 *A Decomposable Problem*

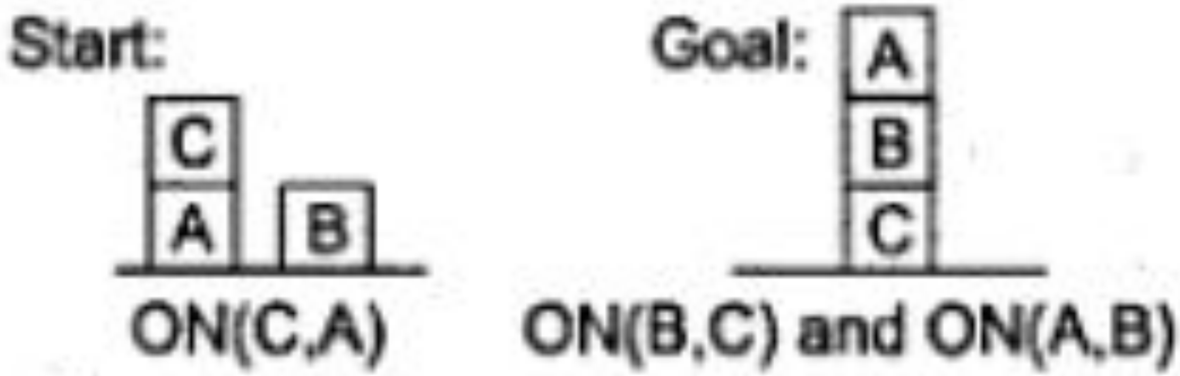


Fig. 2.10 *A Simple Blocks World Problem*

2

Can solution steps be ignored or at least undone if they prove unwise?

Start

2	8	3
1	6	4
7		5

Goal

1	2	3
8		4
7	6	5

Fig. 2.12 *An Example of the 8-Puzzle*

These three problems – theorem proving, 8-puzzle and chess – illustrate the differences between three important classes of problems:

- **Ignorable** (eg theorem proving), in which solution steps can be ignored.
- **Recoverable** (eg. 8-puzzle), in which solution steps can be undone.
- **Irrecoverable** (eg. Chess), in which solution steps cannot be undone.

3

Is the problem's universe predictable?

Certain Outcome Problem

Uncertain Outcome Problem

4

Is a good solution to the problem obvious without comparison to all other possible solutions?

- This means – absolute or relative solution?

1. Marcus was a man.
2. Marcus was a Pompeian.
3. Marcus was born in 40 A.D.
4. All men are mortal.
5. All Pompeians died when the volcano erupted in 79 A.D.
6. No mortal lives longer than 150 years.
7. It is now 1991 A.D.

Is Marcus alive? ------ answer this question?

		Justification
1.	Marcus was a man.	axiom 1
4.	All men are mortal.	axiom 4
8.	Marcus is mortal.	1, 4
3.	Marcus was born in 40 A.D.	axiom 3
7.	It is now 1991 A.D.	axiom 7
9.	Marcus' age is 1951 years.	3, 7
6.	No mortal lives longer than 150 years.	axiom 6
10.	Marcus is dead.	8, 6, 9
	OR	
7.	It is now 1991 A.D.	axiom 7
5.	All Pompeians died in 79 A.D.	axiom 5
11.	All Pompeians are dead now.	7, 5
2.	Marcus was a Pompeian.	axiom 2
12.	Marcus is dead.	11, 2

Fig. 2.13 *Two Ways of Deciding That Marcus Is Dead*

	Boston	New York	Miami	Dallas	S.F.
Boston		250	1450	1700	3000
New York	250		1200	1500	2900
Miami	1450	1200		1600	3300
Dallas	1700	1500	1600		1700
S.F.	3000	2900	3300	1700	

Fig. 2.14 *An Instance of the Traveling Salesman Problem*

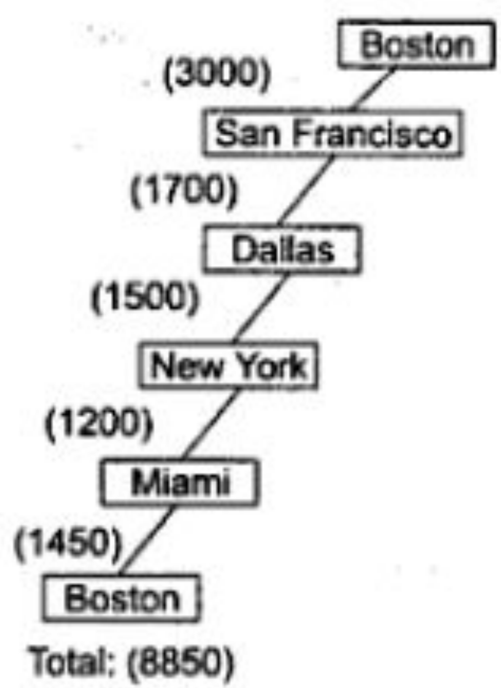


Fig. 2.15 *One Path among the Cities*

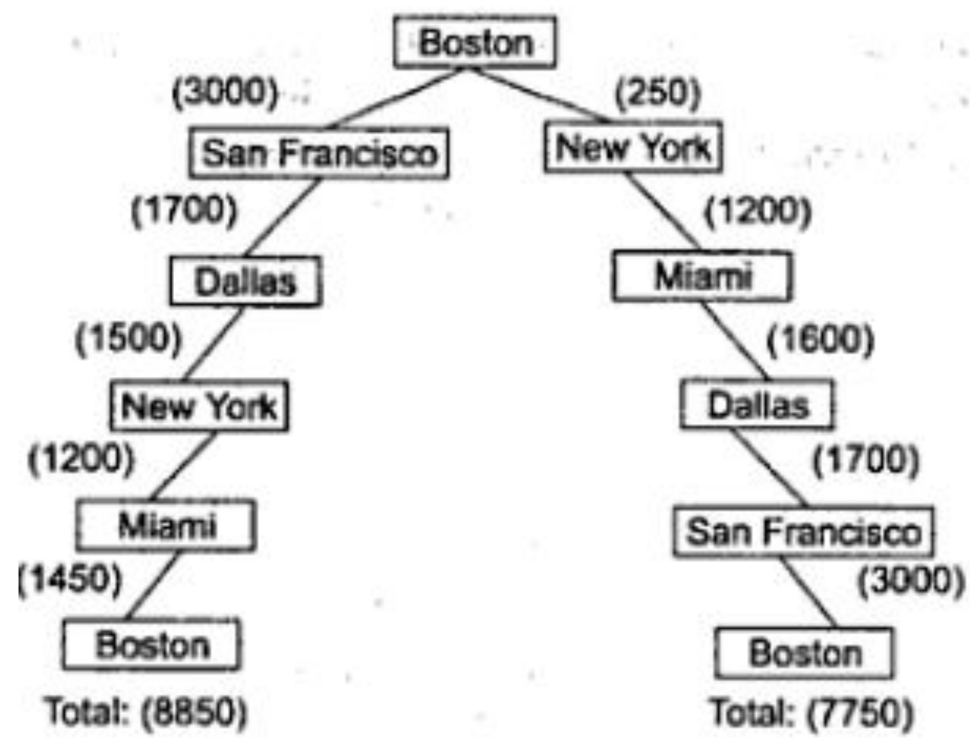


Fig. 2.16 *Two Paths Among the Cities*

5

Is the desired solution a state of the world or a path to a state?

- ***Natural Language Understanding***:

"The ***bank president*** ate a **dish** of **pasta salad** with the **fork**."

There are several components of this sentence, each of which, in isolation, may have more than one meaning or interpretation. But the components must form a coherent whole, and so they constrain each other's interpretations.

- ***Water Jug Problem***:

  Here it is not sufficient to report that we have solved the problem and that the final state is (0,2).

  For this kind of problem, what we really must report is not the final state but the path that we found to that state.

  Thus a statement of a solution to this problem must be a sequence of operations that produces the final state.

6

Is a large amount of knowledge absolutely required to solve the problem or is knowledge important only to constrain the search?

- Playing a chess
  - knowledge important only to constrain the search
- Will the person vote for republican or democrats?
  - large amount of knowledge absolutely required

7

Can a computer that is simply given the problem return the solution, or will the solution of the problem require interaction between the computer and a person?

- ***Solitary***, in which the computer is given a problem description and produces an answer with no intermediate communication and with no demand for an explanation of the reasoning process.

- ***Conversational***, in which there is intermediate communication between a person and the computer, either to provide additional assistance to the computer or to provide additional information to the user, or both.

Production System Characteristics (later)

	Monotonic	Nonmonotonic
Partially commutative	Theorem proving	Robot navigation
Not partially commutative	Chemical synthesis	Bridge

Fig. 2.17 *The Four Categories of Production Systems*

Issues in the Design of Search Programs

- The direction in which to conduct the search (***forward versus backward reasoning***). We can search forward thru the state space from the start state to a goal state or we can search backward from the goal.
- How to select applicable rules (***matching***). Production systems typically spend most of their time looking for rules to apply, so it is critical to have efficient procedures for matching rules against states.
- How to represent each node of the search process (the ***knowledge representation problem*** and the frame problem).

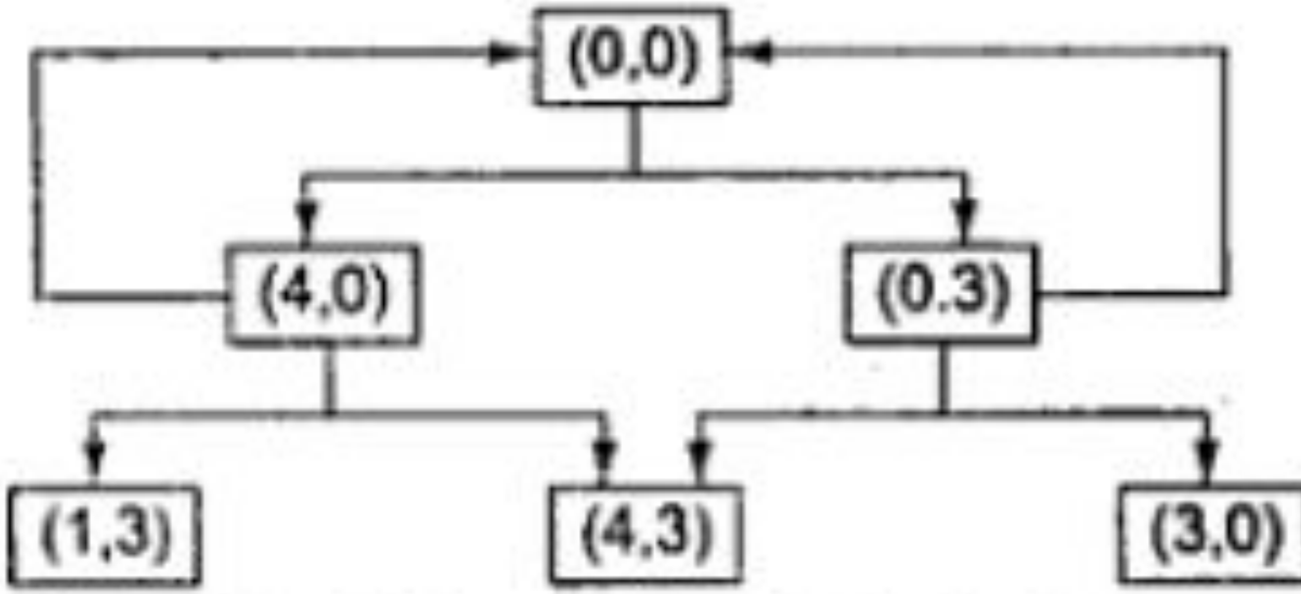
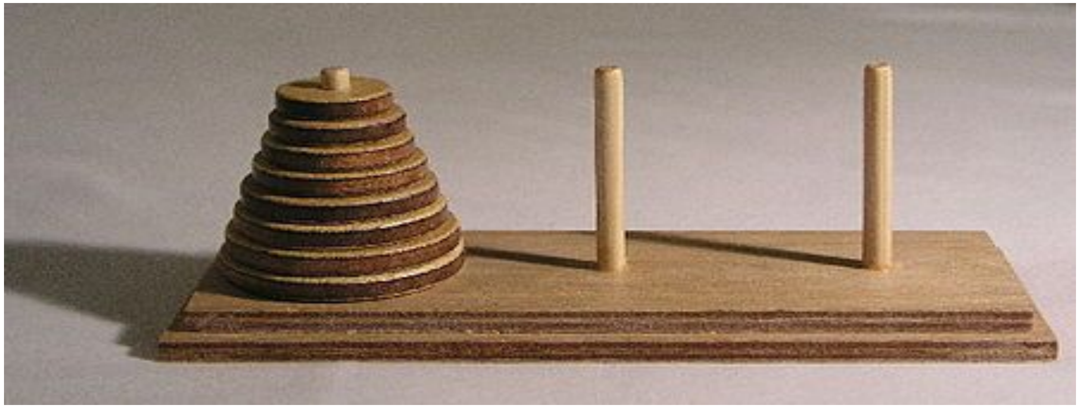


Fig. 2.19 *A Search Graph for the Water Jug Problem*

Additional Problems

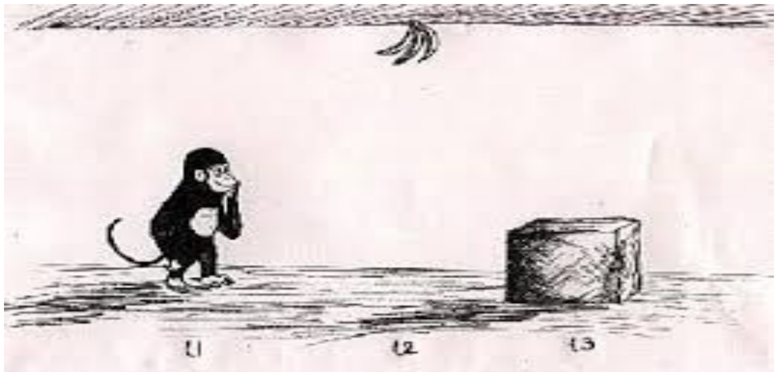
- Missionaries and Cannibals problem

- Tower of Hanoi Problem



Additional Problems

- Monkey and Banana Problem



- Crypt arithmetic Problem

```
 BASE
+BALL
-----
GAMES
```

→

B	7
A	4
S	8
E	3
L	5
G	1
M	9