Sumedh

- Topics

Concept based :-

1. AI- techniques
2. Problem solving with AI
3. AI Models
4. Problem solving
5. Agents
6. Rationality
7. Task environment
8. Agent types

under this

Scenario Based :-

1. Vaccum Cleaner World
2. 8 puzzle / Sliding block
3. Water -jug
4. Queen problem
5. Block world
6. Tic Tac Toe
7. Missionaries & Cannibal
8. Travelling Salesman problem
9. Monkey banana

- AI - Techniques

AI - Techiques are ways to make machine gain knowledge & Solve problem

The Main AI- techniques are :-

1. Search
2. Use of knowledge
3. Abstraction

- Search

- Search is used when no direct method is available to solve a problem
- It tries diff actions until it finds a solution

Adv : i) best method when no other option
ii) you only need to define allowed action , it'll find sol^n itself

disadv : i) No of possible sol^n can be very large

- Use of knowledge

AI uses knowledge to solve complex problems by orgnizing related info

Knowledge shld be structured in a way that :-
  i) grp similar situation in single storage
  ii) easily understood by humans
  iii) Can be updated
  iv) Helps limit unnecessary possibilities to making decision making easier

- Abstraction

  - Ab. means focusing on imp details and ignoring unnecessary ones

  - This helps AI process info efficiently w/o getting confused

- Summary

  1. Knowledge shld be generalized
  2. It shld be easy to update
  3. Shld be usable in diff situations
  4. AI techniques improve problem - solving by inc clarity & efficiency

- Problem solving with AI

AI can be used to solve puzzles and logical problems, such as :

  - suduko
  - 8-queen prob
  - Travelling saleman
  - Water-jug
  - etc

- Problem solving process in AI

To solve probs using AI , follow these :-

i) define the problem
ii) Analyse the problem
iii) Identify possible solⁿ
iv) Choose the best solⁿ
v) Implement the solⁿ

- Types of AI problems

1. Structured

- Well - Structured : have a clear solⁿ
- Ill - structured : No single correct solⁿ

2. Unstructured

- Hard to define
- Often ambiguous

3. Linear

- Have a straight - fwd solⁿ
- eg Classificatⁿ problems

4. Non - linear

- The relationship b/w input & output is complex, making decisions hard

- AI Models

AI models are diff ways of Structuring AI to solve problems, some key models include :-

i) Maze Hypothesis (dunker) : A model that represent AI probs like a maze to find solⁿ

ii) Logic Theory Machine : modes based on logical rules

iii) Natural Lang processing (NLP) : Allow AI to communicate with human

iv) Psychological Models : inspo by human thinking

- Types

1. Semiotic

└ focus on how AI understands symbols, sounds, & words

2. Statistical

└ Use probablity & stats to predict

- data acquisition & learning in AI

AI learns by collecting & analysing data like :-

i) dat mining & ML : find patterns
ii) Computational Learning theory (COLT) : Study how AI learns
iii) Neural Computatn : use brain like models
iv) Intelligent agents : AI that interact with envo
v) Multi-agent system : multiple ai agent working together

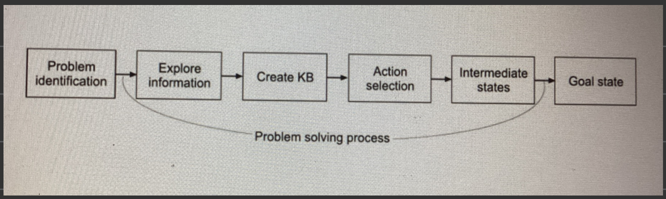
Problem - Solving

It involves finding solution for unknown situations. It includes

- Understanding
- Representation
- Formulation
- Solving

Probs can be simple or complex & are generally solved using two methods :

└ Gen purpose
Special purpose



Problem Solving with AI

Approach : Formulate , Search , Execute

- The agent formulate a goal n a prob
- It searches for a seq
- It executes the action
- Upon completion , it formulates new goal

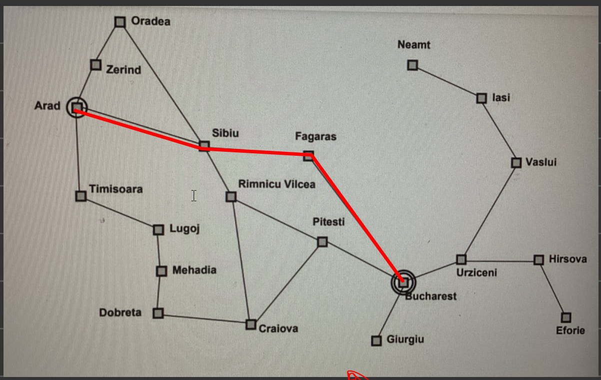
eg : Travelling in Romania

Scenario : On holiday in Romania , currently in arad , flight leaves tom from Bucharest

Goal : Reach Bucharest

Prob formulat^n : States – various cities
Act^n - driv b/w cities

Sol^n :



- Components of a problem

1. Initial State : agent's starting point

eg Arad

2. Possible Act^n : defines available moves

eg All nearest points to Arad

3. Transit^n Model : defines effect of each action

eg : results of those points

4. State space : The set of all possible states reachable from initial state

eg : It forms a directed graph, where nodes are states & linked

5. Goal test : determines whether given state is a goal

eg the goal is Bucharest

6. Path Cost funct^n : Assign a cost to each path

eg dist in kms , the sum determines optimal sol^n

- Problem formulation

Formulation : Choosing relevant states and feasible operators to transit^n b/w states

Search : Imagining seq of operators applied to an initial state to determine which reaches the goal

- Problem types

1. Deterministic

- Single State , Observable
- Each state is fully "
- Transitions definitively
- No Uncertainty involved

eg : A vaccum cleaner with a sensor

2. Non - Observable

- Multiple state, Conformant problems
- The agent lack info
- Sol^n may / may not be reached

eg Vaccum cleaner w/o sensor that blindly suck dirt in all direct^n

3. Non- deterministic

- Partially Observable
- Percepts provide info abt current state

eg A vaccum clear with a sensor that detects dirt before taking act^n

4. Unknown State space

- Typically involves explorat^n prob
- Neither states nor effects of actions are known in adv

eg Online srch w/o prior knowledg

- Problem Characteristics

1. Is the prob decomposable ?

   eg Block world prob

2. Can sol^n steps be ignored or undone ?

   eg 8- puzzle problem

More types : Ignorable – thm proving
Recoverable – 8 puzzle
Irrecoverable - Chess

3. Is the Universe predictable

- Certain Outcome
  eg 8-puzzle
- Uncertain
  eg playing bridge

4. Is a good solⁿ absolute or relative

- Absolute : fixed correct solⁿ
  eg predicate logic
- Relative : Multiple valid solⁿ

5. Is the solⁿ state or a path

- State-based : Interpretatⁿ prob
  eg "He ate food with a fork"
- Path-based : req. tracking steps taken to reach final state

6. What is the role of knowledge

- determines whether extensive knowledge is required to solve the problem or merely constraints the solⁿ

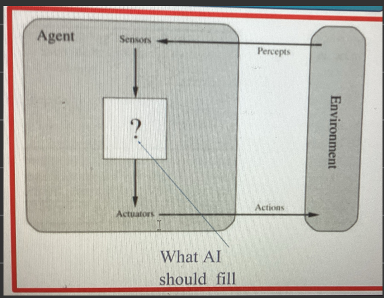
eg i) Knowledge constraing srch
ii) " req. for solⁿ recognition

7. Does the task req. interactⁿ with a person

- Solitary probs : the computer solves prob indep
  eg Chess AI
- Controversial prob : requires intermediate human interaction
  eg AI assisted medical diagnosis

- Agents

An agent is anything that can perceive its envo thru sensors and act upon that envo thru actuators



- Intelligent agent

  L It's an entity that works independantly, interprets inputs, sense the envo, makes decision, act to achieve a goal

Types

1. Human agent : Sensors (eyes, ears, organs)
   actuators (hands, legs, mouth)

2. Robotic : sensors (Cameras, etc)
   actuator (motors)

3. Software agent :-

   - the envo may not always be real world
   - eg flight simulator

- Agent + relationship relationship

Percept : Agent's sensory input at any given moment

Percept seq : The complete history of everthing that agent has perceived

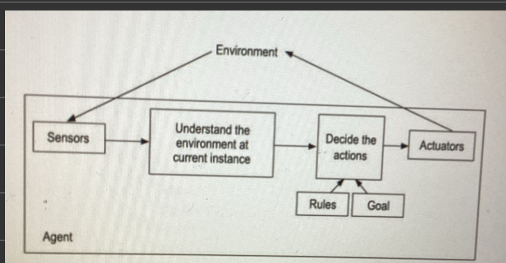


Figure 5.2 Agent-environment relationship.

Maps percept histories to actions : $f : P^* \rightarrow A$

The agent prog runs on a physical arch to execute this func

Agent = Architecture + Prog

eg : Auto-door system

- Use 2 cams (in & out) to monitor

- Simple agent functⁿ :-
  if (area == empty)
  close door
  if (area == occupied)
  open door

- The agent prog ensures
  Continuous scanning
  proper door control
  Handling all edge cases

- **Rationality**

Rationality is the ability to make reasonable, logical & well-judged decisions based on perceived info

- Characteristics

i) Acts logically & does the right thing
ii) Chooses actions that led to optimal outcomes
iii) Use all available knowledge & perceptual inputs
iv) Takes rational actions

eg : Humans, govt, org, machines, s/w

- **Ideal Rational Agent**

An ideal Rational agent is the one that maximises performance based on :
i) Buit-in knowledge base
ii) performance measure

iii) Available actions
iv) percept seq

- What defines a rational actⁿ
  - It's the one that measures highest possible success
  - Uses PEAS
    - (Performance measure
      Environment
      Actuators
      Sensors) to define problem scope

- Rational depends on 4 things :

1. performance measure
2. percept seq
3. Knowledge to envo
4. Actⁿ cappabilities

eg : Automated Taxi driver

Performance Measure : Safe , fast , legal , max profits

Envo : roads , traffic

Actuators : Steering , acceleration

Sensors : Cams , sonar , GPS

- Concept of Rationality
  - How to Measure ?

- Performance measure : defines the success lvl
- exp performance vs Actual performance :-
  (rational agent maximise expected performance)
  (perfect agent maximise actual performance)

- Omniscience

  └ this agent knows all possible outcome beforehand

- No other possible outcomes
- Impossible in real world

eg : A rational agent crossing the street may still face unpredictable dangers (like falling cargo door frm 1000 ft.)

- Learning

  - A RA shld learn frm past exp to improve future decisions
  - If an agent only follows pre-programmed rules, it lacks autonomy

eg A clock , runs on internal mechanism , no percepts & adapt"

- Task environment

  └ The problem the agent is solving

- Uses PEAS

  └ P : defines success criteria
  E : the world that agent interacts with
  A : The control the agent has
  S : devices for percept"

eg : Automated taxi driver

└ (" same as last pg ")

· Properties of task envo

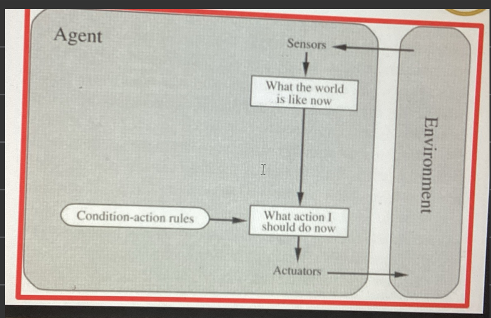
Property	Description	Example
Fully Observable	The agent can perceive the complete environment.	Chess
Partially Observable	Some information is missing or inaccurate.	Taxi driving
Deterministic	Next state is fully determined by current state & action.	Chess
Stochastic	The next state is uncertain due to randomness.	Taxi driving
Episodic	Actions are independent of past actions.	Image recognition
Sequential	Past actions affect future actions.	Chess, Taxi driving
Static	The environment does not change unless the agent acts.	Crossword puzzle
Dynamic	The environment keeps changing.	Driving
Discrete	Limited states and actions.	Chess
Continuous	Infinite states and actions.	Self-driving cars
Single-Agent	One agent operating in the environment.	Crossword puzzle
Multi-Agent	Multiple agents interact.	Chess, taxi driving
Known	The agent knows all possible actions and outcomes.	Solitaire
Unknown	The agent must learn through experience.	New video game

• Agent types

· Simple reflex Agents

- Uses 'if - then' rules
- efficient
- Only works in fully observable envo

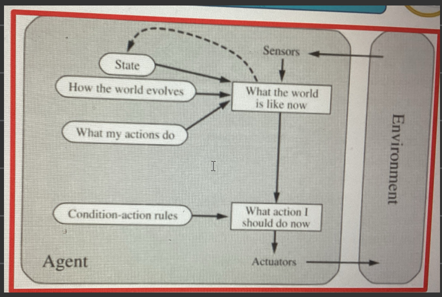
eg : A frog snaps at small obj avoiding large ones



- Model-based reflex agent

  - Maintains an internal state to track past percepts
  - Useful in partially observable env

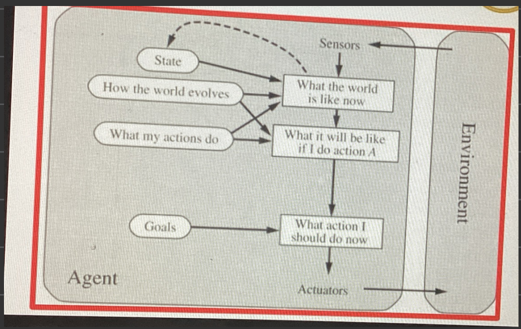
eg a car maintains memory of past lane changes



- Goal-based agents

  - Chooses actions based on achieving a specific goal
  - Req. search & planning
  - More flexible

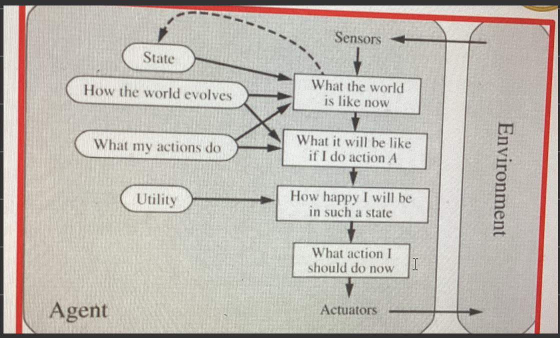
eg g map finding the shortest route



- Utility-based agents

  - Uses a utility funct^n to measure the desireability of states
  - Can make trade-offs b/w conflicting goals

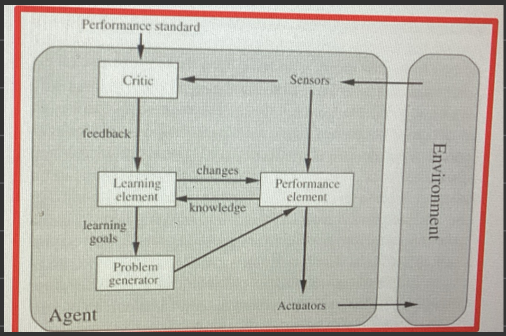
eg : An ai recommending personalized content based on user preference



• Learning agents

- learning elem : improves the agent's performance
- performance elem : select act^n
- Critic : evaluates the agent's performance
- prob generator : suggests new exp for learning

eg AI - powered self - driving cars



- **Toy problems** (* In exam scenario will be given with value, write algo acc. to those n find final ans)

1. **Vacuum Cleaner World**

Imagine you have a small room with 2 sectⁿ and a robot vacuum cleaner that moves b/w them.

The floor may have dirt and it's job is to clean it

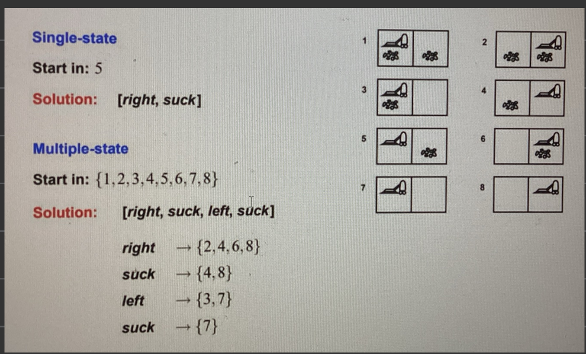
State : the vac clnr can either be in 1/2 loc n that loc may be clear or dirty

i.e $2^2 \times 2 = 8$ possibilities

Initial State : The vacuum starts in any of these 8 states

Action : move left or rgt to suck n clear

eg :



(* use BFS / DFS approach)

2. **8 - puzzle / Sliding block**

This puzzle consists of a 3×3 grid with 8 no. tiles and one empty space. You can move a tile into empty space. The goal is to arrange the tiles in a specific order

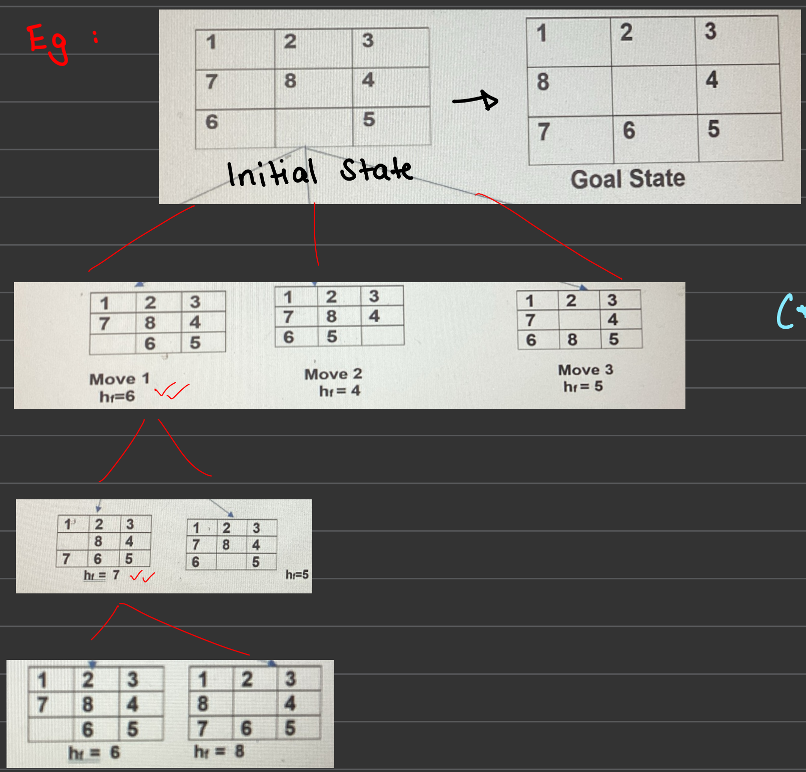
State : A state represents the pos of all the tiles & the empty space

Initial state : Any arrangement

Actions : Use heuristic approach, move the blank space left, rgt, up, down

Approach : Heuristic approach

Eg :



↳ Go to all possible dir then check how many at correct place 'hr ++', proceed with the next step with high 'hr'

3. Water - Jug problem

Here 2 jugs of diff vol will be given (v_1, v_2), intial state will be given in order to find final

Approach : i) Hit n trail
ii) DFS

eg : 2 jugs, 5 gallon n 3 gallon
ini (0, 0)
final (4, 0)

i) Hit n trail method

5G 3G
(0, 0)
↓
(0, 3) fill 3G one
↓
(3, 0) pour in 5G
↓
(3, 3) fill 3G again
↓
(5, 1) pour in 5G again, 3G → 1G left
|

$\downarrow$

(0, 1) empty 5G

$\downarrow$

(1, 0) pour 1 to 5G

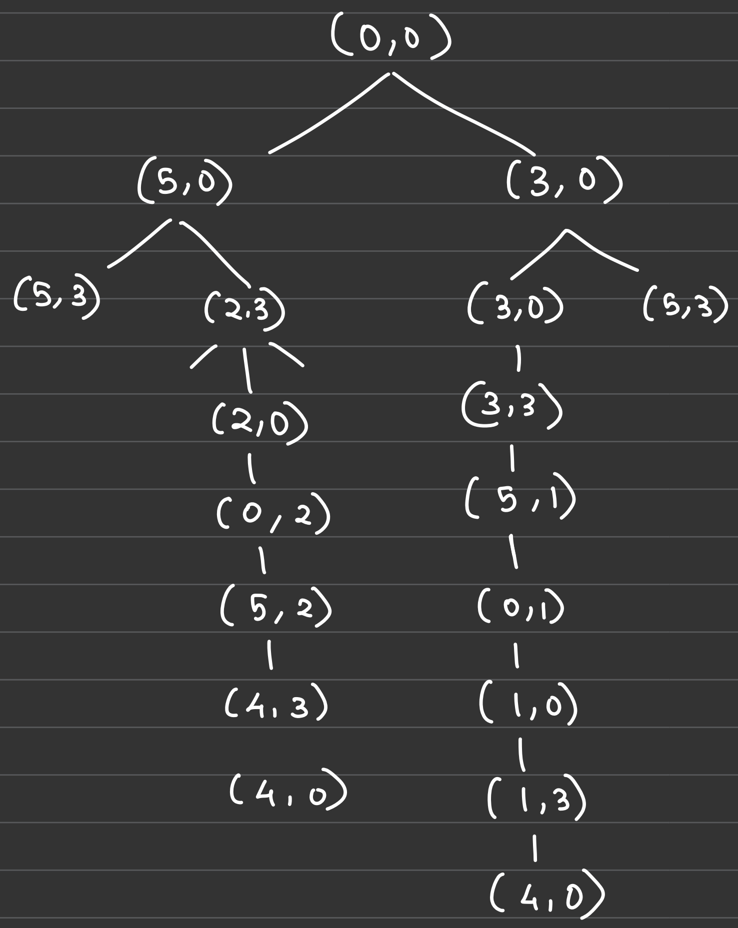
$\downarrow$

(1, 3) fill 3G again

$\downarrow$

✓✓ (4, 0) pour it in 5G

ii) DFS



4. n queen prob

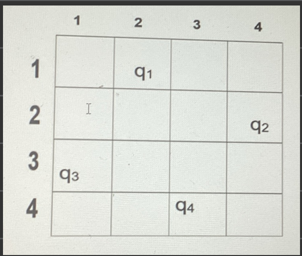
placing n queen in a n×n chessboard so no 2 queens attack each other i.e lft, rgt, diagonal

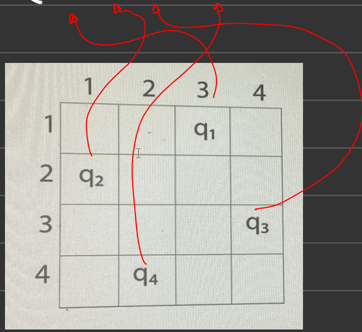
eg 4-queen, 8-queen

Approach: Backtracking (place one by one & backtrack when conflict

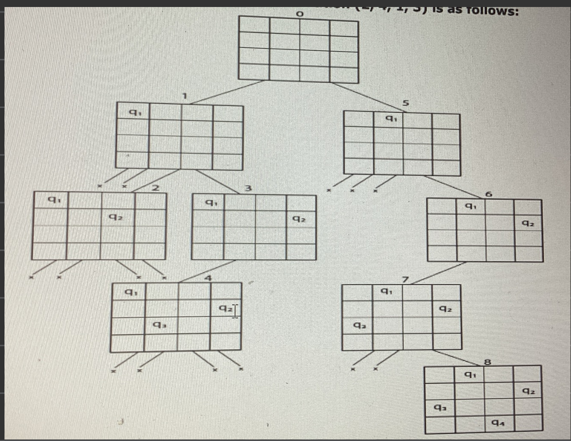
* for 4-queen only 2 possible sol^n

(2, 4, 1, 3) & (3, 1, 4, 2)

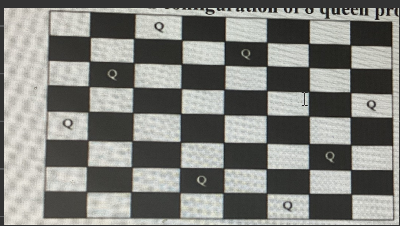




eg :



eg 8-queen



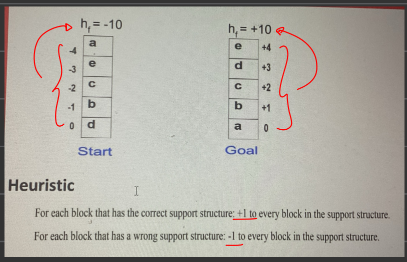
5. Block World

It's a puzzle where blocks are stacked on a table, a robot arm can pick up, move & stack block

Constraints : i) only one block on another
ii) any no. of blocks on table
iii) arm can only hold one block

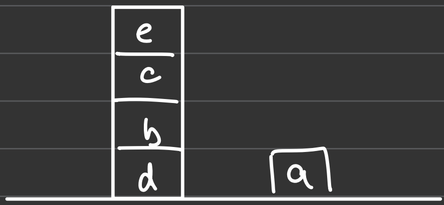
Approach : Heuristic

eg :



S-1

hr = -6



S-2

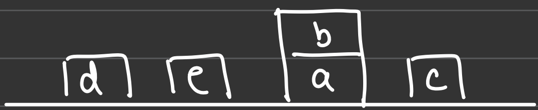
hr = -3

S-3

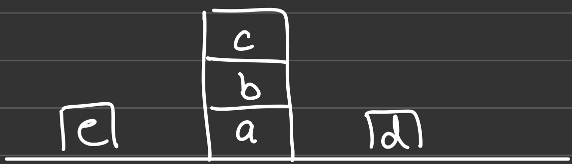
hr = -1

S-4

hr = +1



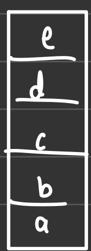
S-5



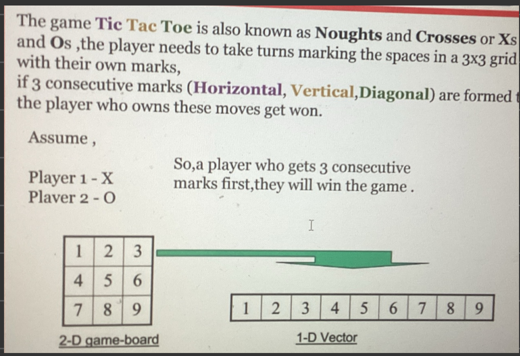
S-6

S-7

hr = +6



6. Tic - Tac - Toe


The game Tic Tac Toe is also known as Noughts and Crosses or Xs and Os ,the player needs to take turns marking the spaces in a 3x3 grid with their own marks,
if 3 consecutive marks (Horizontal, Vertical,Diagonal) are formed t the player who owns these moves get won.

Assume ,

Player 1 - X
Player 2 - O

So,a player who gets 3 consecutive marks first,they will win the game .

7. Missionaries & Cannibals (M n C)

Problem statement :-

- You have 3M & 3C on one side (bank-1) of a river
- one boat that can carry at max 2
- Goal to get everyone on other side (bank-2) safely

Rules :
- If on any side M < C , C eats M you loose
- boat cannot go empty , shld atleast take one

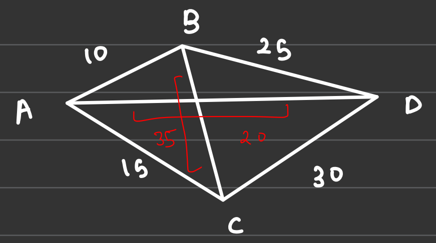
Solving :-

Bank -1	Bank-2	river
M M M C C C	0	bank-1
M M C C	M C	2
M M M C C	C	1
M M M	C C C	2
M M M C	C C	1
M C	M M C C	2
M C C M	M C	1
C C	M M M C	2
C C C	M M M C	1
C	M M M C C	2
C C	M M M C	1
0	M M M C C C	2

8. Travelling Salesman problem (TSP)

Here the salesman needs to travel from cities to a goal city with shortest dist

eg



Initial : A

Goal : return A

i) Brute force : Calculate all possible n given one with shortest

1. A → B → C → D → A
   - Distance: 10 + 35 + 30 + 20 = 95
2. A → B → D → C → A
   - Distance: 10 + 25 + 30 + 15 = 80
3. A → C → B → D → A
   - Distance: 15 + 35 + 25 + 20 = 95
4. A → C → D → B → A
   - Distance: 15 + 30 + 25 + 10 = 80
5. A → D → B → C → A
   - Distance: 20 + 25 + 35 + 15 = 95
6. A → D → C → B → A
   - Distance: 20 + 30 + 35 + 10 = 95

∴ Optimal

TC = $O(n!)$

better

ii) Greedy approach : go for nearest neighbour

or

Heuristic

A → B (10)

A → B → D (25)

TC = $O(n^2)$

A → B → D → C (30)

A → B → D → C → A (15)

Total = 80

9. Monkey - Banana problem

A monkey is in a room where a banana is hanging from the ceiling, the monkey wants to grap the banana using table

eg : Monkey at (1,1) (M)
Chair at (2,2) (C)
Banana at (3,3) (B)

Initial State : M (1,1) , C (2,2), B (3,3)

S-1 : M walks to (2,2)
State : M (2,2) , C (2,2) , B (3,3)

S-2 : M push C to (3,3)
State : M (3,3) , C(3,3) , B (3,3)

S-3 : M climbs C at (3,3)
State : M , C, B (3,3)

S-4 : M eats B !