ITC703

Artificial Intelligence

1. Introduction to Intelligent Systems and Intelligent Agents

- Introduction to AI, AI Problems and AI techniques,
- Solving problems by searching, Problem Formulation.
- State Space Representation Structure of Intelligent agents,
- Types of Agents, Agent Environments PEAS representation for an Agent.

2. Search Techniques

- Uninformed Search: DFS, BFS, Uniform cost search, Depth Limited Search, Iterative Deepening.
- Informed Search : Heuristic functions, Hill Climbing, Simulated Annealing, Best First Search, A*, Constraint Satisfaction Programming: Crypto Arithmetic, Map Coloring, N-Queens.
- Adversarial Search: Game Playing, Min-Max Search, Alpha Beta Pruning

3. Knowledge and Reasoning

- A Knowledge Based Agent, Overview of Propositional Logic, First Order Predicate Logic, Inference in First Order
- Predicate Logic: Forward and Backward Chaining, Resolution.

4. Planning

- Introduction to Planning, Planning with State Space Search,
- Partial Ordered planning, Hierarchical Planning, Conditional Planning.

5. Uncertain Knowledge and Reasoning

- Uncertainly, Representing Knowledge in an Uncertain Domain, Conditional Probability, Joint Probability, Bayes' theorem, Belief Networks, Simple Inference in Belief Networks.

6. Natural Language Processing

- Language Models, Natural Language for Communication : Syntactic Analysis, Augmented Grammars and Semantic Interpretation, Machine Translation.
- **Overview of Cognitive Computing :** Foundation of Cognitive Computing, List of Design Principles for Cognitive Systems, Natural Language Processing in Support of a Cognitive System

Mini project

- Construction of a domain specific ChatBot using Natural Language Processing techniques.
- (Applications can include : Medical Diagnosis, Personal Shopping Assistant, Travel Agent , Trouble shooting etc.)

Text Books

- 1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 2nd Edition, Pearson Education.
- 2. Elaine Rich, Kevin Knight, Shivshankar B Nair, Artificial Intelligence, McGraw Hill, 3rd Edition
- 3. Judith S. Hurwitz, Marcia Kaufman, Adrian Bowles, Cognitive Computing and Big Data Analytics, Wiley India
- Deepak Khemani, A First Course in Artificial Intelligence, McGraw Hill Publication

Introduction to Artificial Intelligence

John McCarthy who has coined the word “Artificial Intelligence” in 1956, has defined AI as “the science and engineering of making intelligent machines”, especially intelligent computer programs.

Artificial Intelligence (AI) is relevant to any intellectual task where the machine needs to take some decision or choose the next action based on the current state of the system, in short act intelligently or rationally. As it has a very wide range of applications, it is truly a universal field.

Introduction to Artificial Intelligence

- In simple words, Artificial Intelligent System works like a Human Brain, where a machine or software shows intelligence while performing given tasks; such systems are called **intelligent systems** or **expert systems**. You can say that these systems can “think” while generating output!!!

- AI is the study of how to make machines do thing which at the moment people do better. Following are the four approaches to define AI.

Four Approaches to define AI

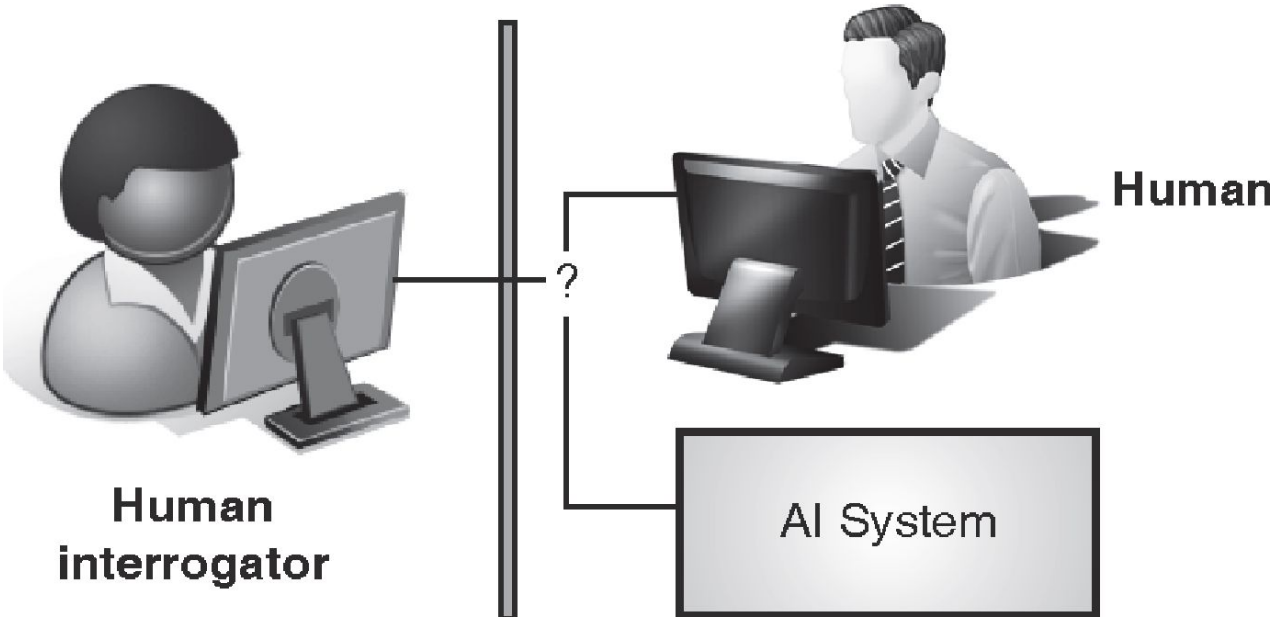
- Acting Humanly : The Turing Test Approach
- Thinking Humanly : The Cognitive Modelling Approach
- Thinking Rationally : The “Laws of Thought” Approach
- Acting Rationally : The Rational Agent Approach

Acting Humanly : The Turing Test Approach

- **Definition 1 :** “The art of creating machines that perform functions that requires intelligence when performed by people.” (Kurzweil, 1990)

- **Definition 2 :** “The study of how to make computers do things at which, at the moment, people are better.” (Rich and Knight, 1991)

Turing Test Environment



For this test, the computer would need to possess the following capabilities

1. **Natural Language Processing (NLP) :** This unit enables computer to interpret the English language and communicate successfully.

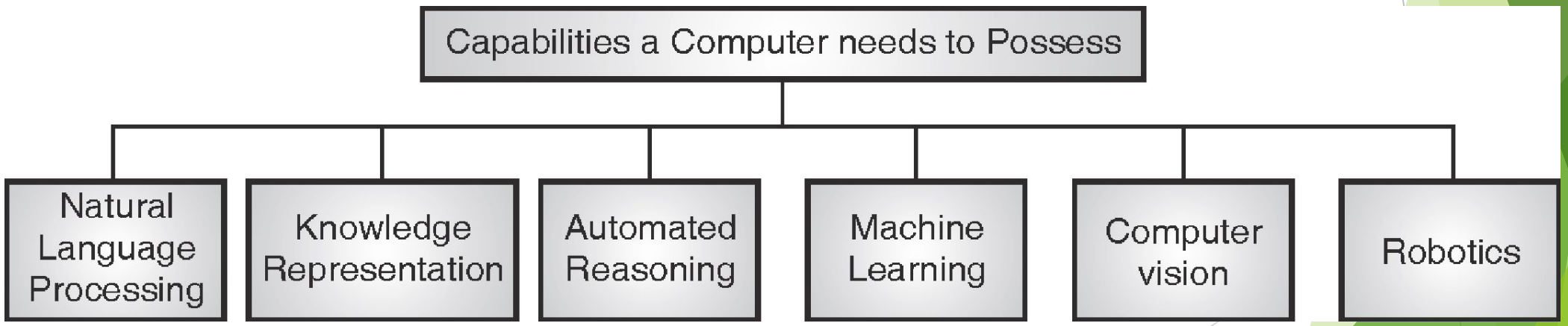
2. **Knowledge Representation :** This unit is used to store knowledge gathered by the system through input devices.

3. **Automated Reasoning:** This unit enables to analyze the knowledge stored in the system and makes new inferences to answer questions.

4. **Machine Learning:** This unit learns new knowledge by taking current input from the environment and adapts to new circumstances, thereby enhancing the knowledgebase of the system.

Turing Test

- To pass total Turing test, the computer will also need to have **computer vision,** which is required to perceive objects from the environment and **Robotics,** to manipulate those objects.



Thinking Humanly : The Cognitive Modelling Approach

Definition1 : “The exciting new effort to make computers think ... machines with minds, in the full and literal sense”. (Haugeland, 1985)

Definition 2 : “The automation of activities that we associate with human thinking, activities such as decision making, problem solving, learning ...” (Hellman, 1978)

Cognitive science : It is interdisciplinary field which combines computer models from Artificial Intelligence with the techniques from psychology in order to construct precise and testable theories for working of human mind.

Three ways using which human's thinking pattern can be caught

1. **Introspection** through which human can catch their own thoughts as they go by.
2. **Psychological experiments** can be carried out by observing a person in action.
3. **Brain imaging** can be done by observing the brain in action.

By catching the human thinking pattern, it can be implemented in computer system as a program and if the program's input output matches with that of human, then it can be claimed that the system can operate like humans.

Thinking Rationally : The “Laws of Thought” Approach

Definition1 : “The study of mental faculties through the use of computational models”. (Charniak and McDermott, 1985)

Definition2 : “The study of the computations that make it possible to perceive, reason, and act”.

It also includes reasoning and “right thinking” that is irrefutable thinking process. Also computer programs based on those logic notations were developed to create intelligent systems.

There are two problems in this approach :

1. This approach is not suitable to use when 100% knowledge is not available for any problem.
2. As vast number of computations was required even to implement a simple human reasoning process; practically, all problems were not solvable because even problems with just a few hundred facts can exhaust the computational resources of any computer.

Acting Rationally : The Rational Agent Approach

Definition 1 :“Computational Intelligence is the study of the design of intelligent agents”. (Poole et at, 1998)

Definition 2:“Al ... is concerned with intelligent behaviour in artifacts”.(Nilsson, 1998)

Topics Covered in Last Lecture

- Introduction to AI
- Four Approaches to define AI
- Acting Humanly
- Turing Test
- Computer must possess NLP, Knowledge Representation, Automated Reasoning, Machine Learning , Computer Vision and Robotics
- Thinking Humanly
- Thinking Rationally
- Acting Rationally

Rational Agent

- Agents perceive their environment through sensors over a prolonged time period and adapt to change to create and pursue goals and take actions through actuators to achieve those goals.
- A rational agent is the one that does “right” things and acts rationally so as to achieve the best outcome even when there is uncertainty in knowledge.

Advantages of rational-agent approach

1. As compared to other approaches this is the more general approach as, rationality can be achieved by selecting the correct inference from the several available.
2. Rationality has specific standards and is mathematically well defined and completely general and can be used to develop agent designs that achieve it. Human behavior, on the other hand, is very subjective and cannot be proved mathematically.

Categorization of Intelligent Systems

- **Artificial Narrow Intelligence/ Weak AI**
- Weak AI is AI that specializes in one area. It is not a general purpose intelligence. An intelligent agent is built to solve a particular problem or to perform a specific task is termed as narrow intelligence or weak AI.
- For example, it took years of AI development to be able to beat the chess grandmaster, and since then we have not been able to beat the machines at chess.

But that is all it can do, which is does extremely well.

Categorization of Intelligent Systems

- **Artificial General Intelligence / Strong AI**
- Strong AI or general AI refers to intelligence demonstrated by machines in performing any intellectual task that human can perform. Developing strong AI is much harder than developing weak AI.
- Using artificial general intelligence machines can demonstrate human abilities like reasoning, planning, problem solving, comprehending complex ideas, learning from self experiences, etc.

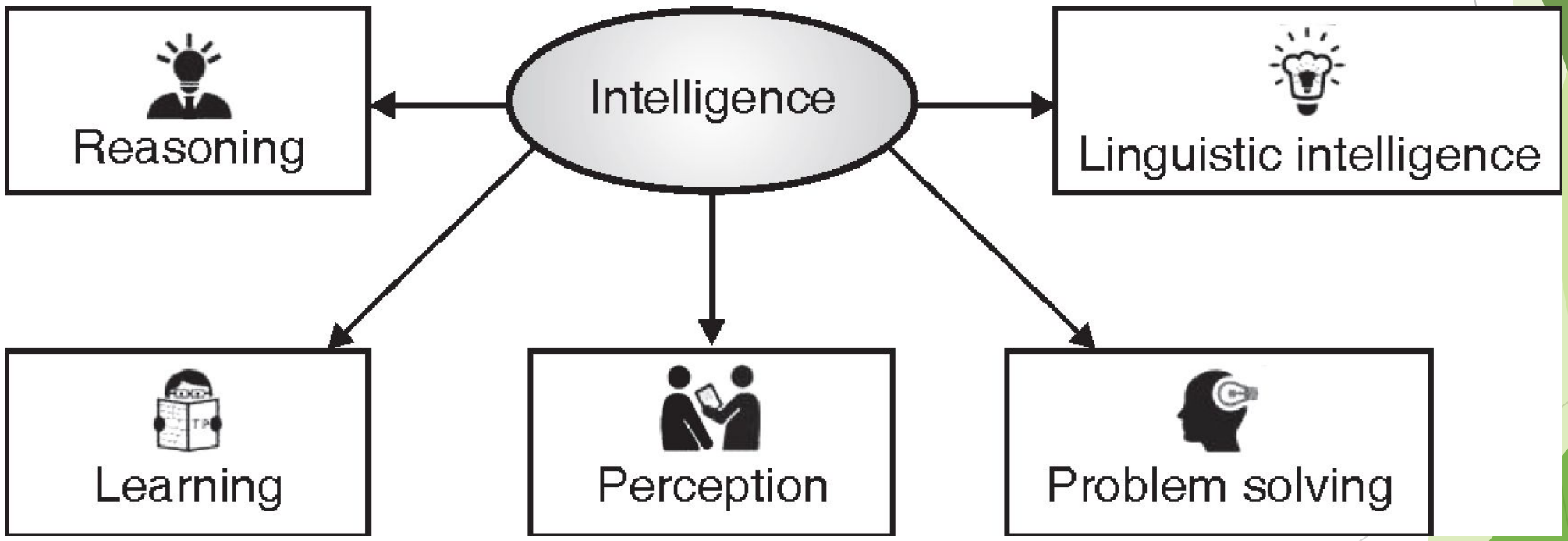
Categorization of Intelligent Systems

- **Artificial Super Intelligence**
- As defined by a leading AI thinker Nick Bostrom, “Super intelligence is an intellect that is much smarter than the best human brains in practically every field, including scientific creativity, general wisdom and social skills.”
- Super intelligence ranges from a machine which is just a little smarter than a human to a machine that is trillion times smarter.
- Artificial super intelligence is the ultimate power of AI.

Components of AI

1. Perception
2. Knowledge representation
3. Learning
4. Reasoning
5. Problem solving
6. Natural language processing (Language-understanding)

Components of AI



Perception

- In order to work in the environment, intelligent agents need to scan the environment and the various objects in it.
- Agent scans the environment using various sense organs like camera, temperature sensor, etc. This is called as perception.
- After capturing various scenes, perceiver analyses the different objects in it and extracts their features and relationships among them.

Knowledge representation

- The information obtained from environment through sensors may not be in the format required by the system.
- Hence, it need to be represented in standard formats for further processing like learning various patterns, deducing inference, comparing with past objects, etc.
- There are various knowledge representation techniques like Prepositional logic and first order logic.

Learning

- Learning is a very essential part of AI and it happens in various forms. The simplest form of learning is by trial and error.
- In this form the program remembers the action that has given desired output and discards the other trial actions and learns by itself. It is also called as **unsupervised learning.**
- In other case, solution to few of the problems is given as input to the system, basis on which the system or program needs to generate solutions for new problems. This is known as **supervised learning.**

Reasoning

- Reasoning is also called as logic or generating inferences form the given set of facts. Reasoning is carried out based on strict rule of validity to perform a specified task. Reasoning can be of two types, deductive or inductive.
- The **deductive reasoning** is in which the truth of the premises guarantees the truth of the conclusion while, in case of **inductive reasoning**, the truth of the premises supports the conclusion, but it cannot be fully dependent on the premises.
- In programming logic generally deductive inferences are used. Reasoning involves drawing inferences that are relevant to the given problem or situation.

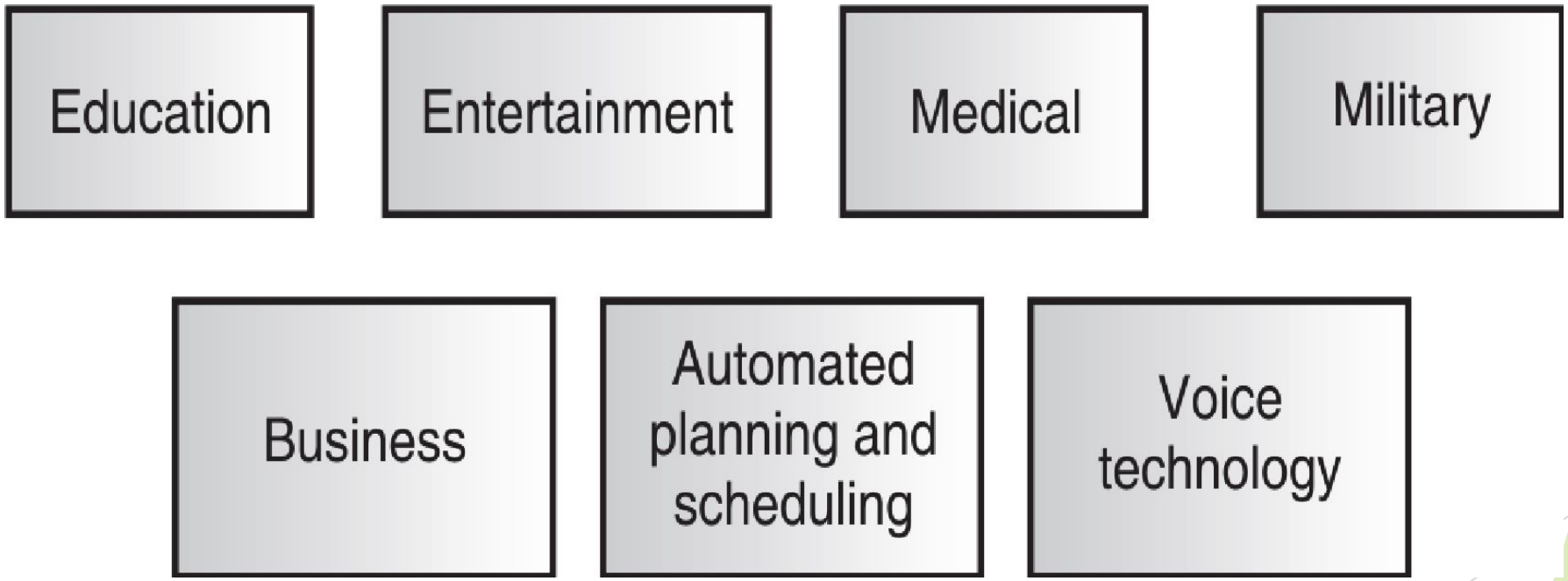
Problem-solving

- AI addresses huge variety of problems. For example, finding out winning moves on the board games, planning actions in order to achieve the defined task, identifying various objects from given images, etc.
- As per the types of problem, there is variety of problem solving strategies in AI. Problem solving methods are mainly divided into **general purpose methods** and **special purpose methods**.
- **General purpose** methods are applicable to wide range of problems while, **special purpose** methods are customized to solve particular type of problems

Natural Language Processing(NLP)

- Natural Language Processing, involves machines or robots to understand and process the language that human speak, and infer knowledge from the speech input. It also involves the active participation from machine in the form of dialog
- i.e. NLP aims at the text or verbal output from the machine or robot. The input and output of an NLP system can be speech and written text respectively.

Applications of Artificial Intelligence



Applications of Artificial Intelligence

1. Education

Training simulators can be built using artificial intelligence techniques. **Software for pre-school children** are developed to enable learning with fun games. **Automated grading, Interactive tutoring, instructional theory** are the current areas of application.

2. Entertainment

Many **movies, games, robots** are designed to play as a character. In games they can play as an opponent when human player is not available or not desirable.

Applications of Artificial Intelligence

3. Medical

- AI has applications in the field of cardiology (CRG), Neurology (MRI), Embryology (Sonography), complex operations of internal organs, etc. It can be also used in **organizing bed schedules**, **managing staff rotations, store and retrieve information of patient**. Many expert systems are enabled to predict the disease and can provide with medical prescriptions.

Applications of Artificial Intelligence

4. **Military**

- **Training simulators** can be used in military applications. Also areas where human cannot reach or in life stacking conditions, robots can be very well used to do the required jobs.
- When **decisions** have to be made **quickly** taking into account an enormous amount of information, and when lives are at stake, artificial intelligence can provide crucial assistance.
- From developing intricate flight plans to implementing complex supply systems or creating training simulation exercises, AI is a natural partner in the modern military.

Applications of Artificial Intelligence

5.Business and Manufacturing

Latest generation of robots are equipped well with the performance advances, growing integration of vision and an enlarging capability to transform manufacturing

6. Automated planning and scheduling

Intelligent planners are available with AI systems, which can process large datasets and can consider all the constraints to design plans satisfying all of them.

Applications of Artificial Intelligence

7.Voice technology

- Voice recognition is improved a lot with AI. Systems are designed to take voice inputs which are very much applicable in case of handicaps. Also scientists are developing an intelligent machine to emulate activities of a skillful musician.
- **Composition, performance, sound processing, music theory are some of the major areas of research.**

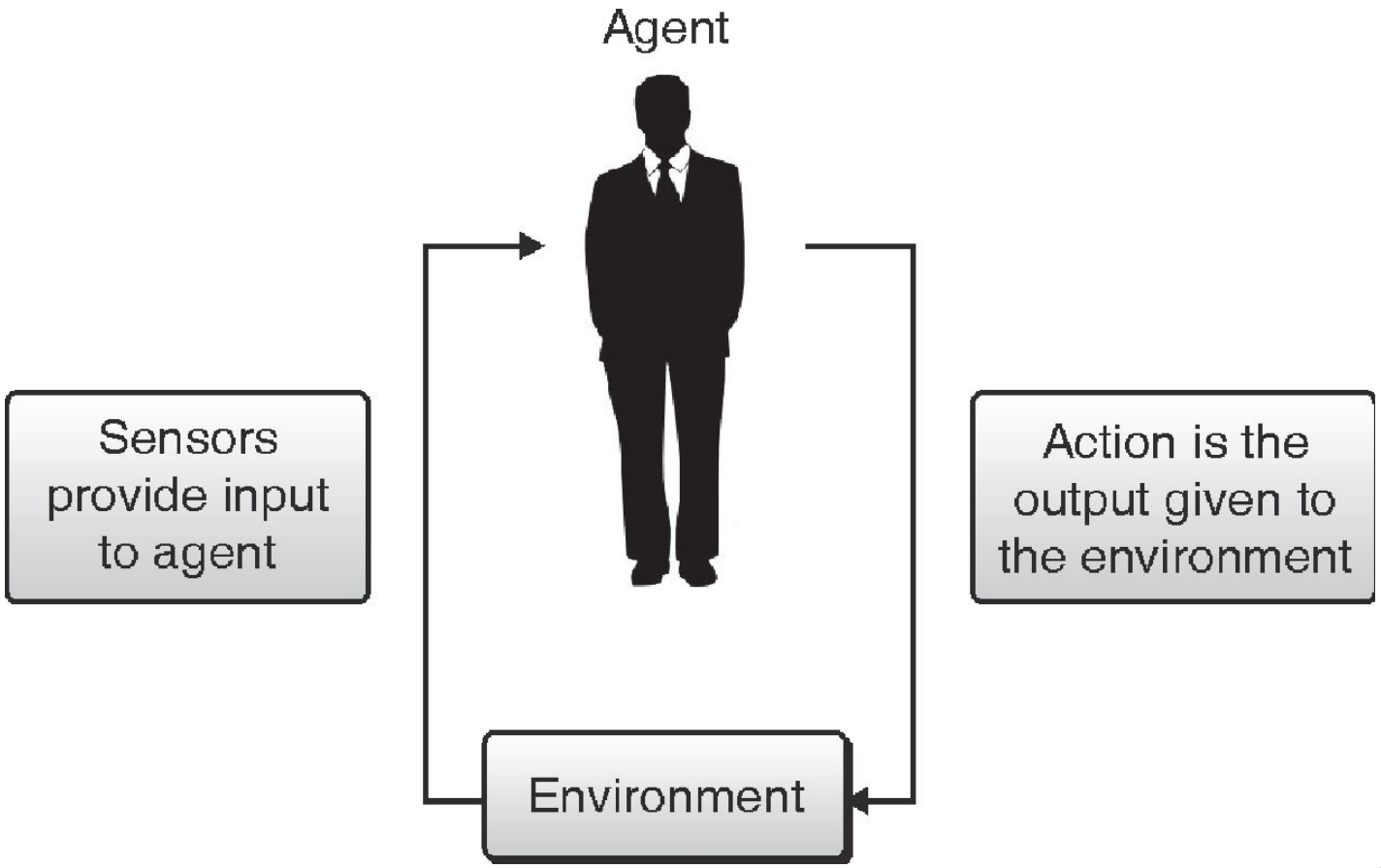
8. Heavy industry

- **Huge machines involve risk in operating and maintaining them. Human robots are better replacing human operators.**
- **These robots are safe and efficient. Robot are proven to be effective as compare to human in the jobs of repetitive nature, human may fail due to lack of continuous attention or laziness.**

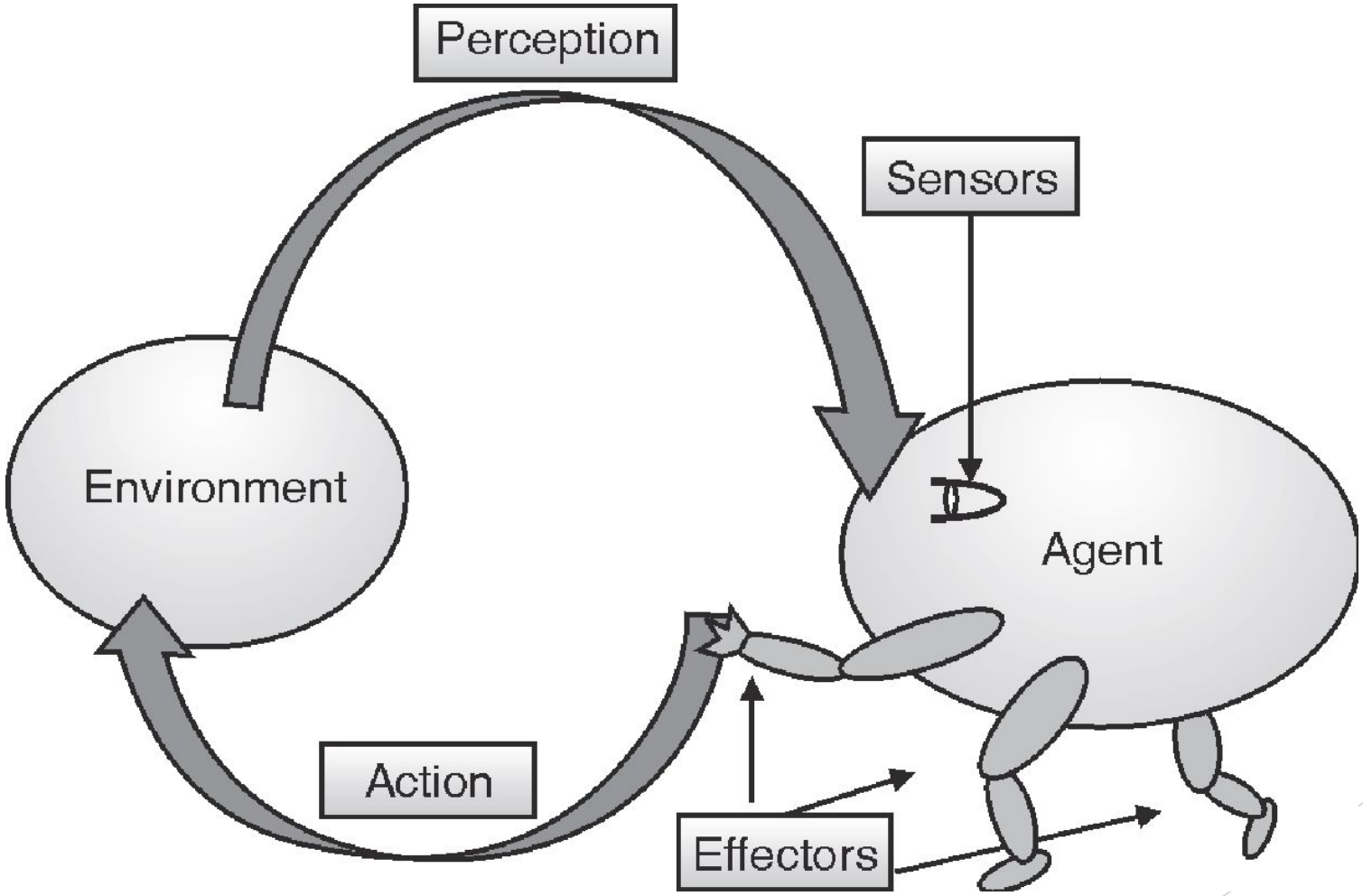
What is Agents?

- **Agent** is something that perceives its environment through sensors and acts upon that environment through effectors or actuators
- Take a simple example of a human agent. It has five senses : Eyes, ears, nose, skin, tongue. These senses sense the environment are called as **sensors**. Sensors collect percepts or inputs from environment and passes it to the processing unit.
- **Actuators or effectors** are the organs or tools using which the agent acts upon the environment. Once the sensor senses the environment, it gives this information to nervous system which takes appropriate action with the help of actuators.

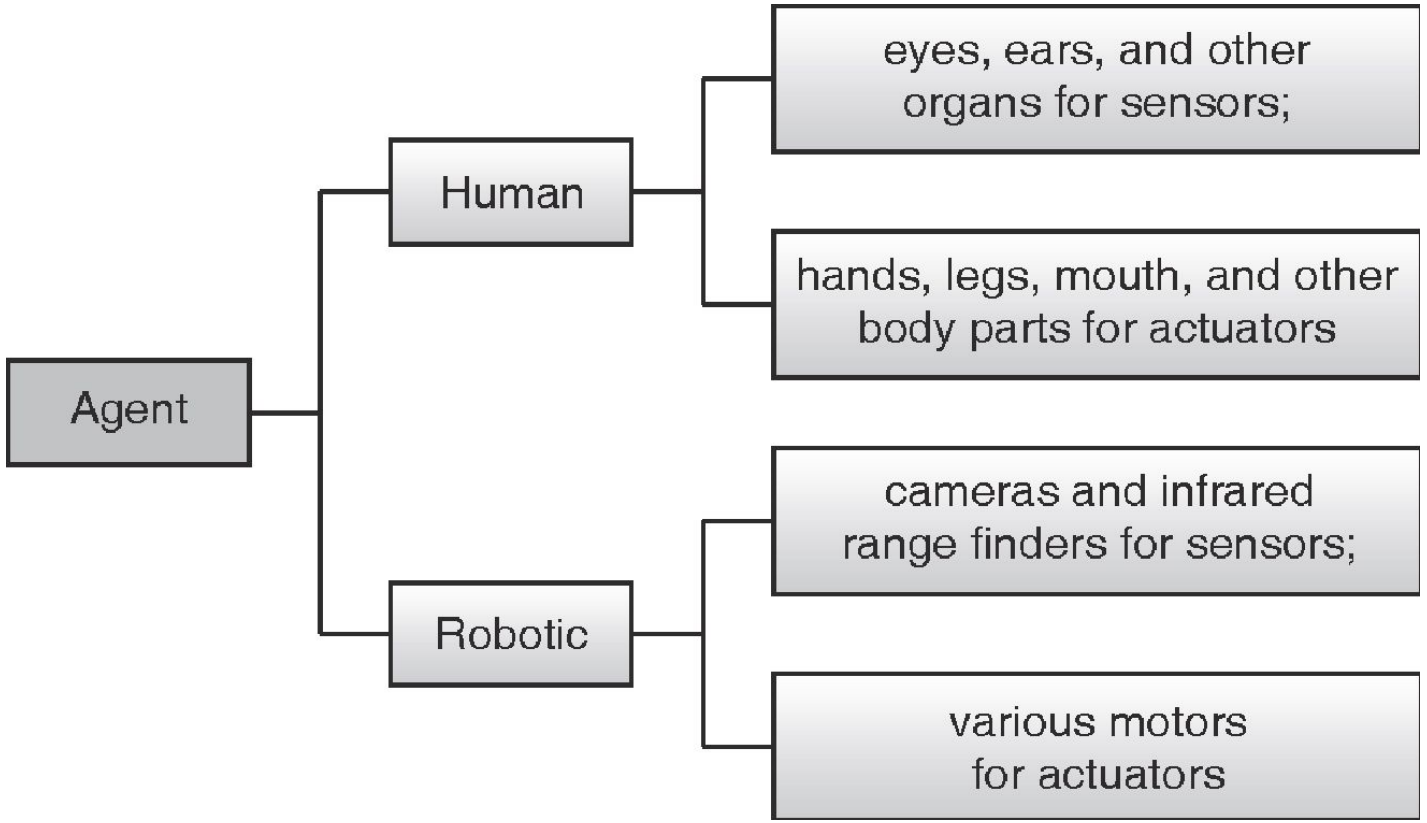
Agent and Environment



Generic robotic agent architecture



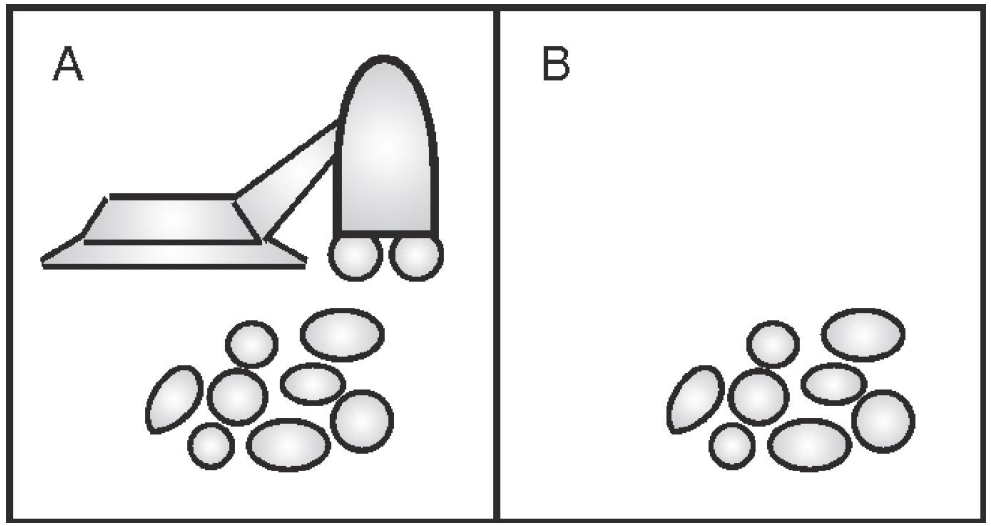
Sensors and actuators in human and robotic agent



Agent program

- **Agent program is a computer program that** implements agent function in an architecture suitable language.
- Agent programs needs to be installed on a device in order to run the device accordingly. That device must have some form of sensors to sense the environment and actuators to act upon it.
- Hence agent is a combination of the architecture hardware and program software.
- **Agent = Architecture + Program**

Vacuum cleaner agent



There are two blocks A and B having some dirt. Vacuum cleaner agent supposed to sense the dirt and collect it, thereby making the room clean.

Vacuum cleaner agent

- Hence the sensor for vacuum cleaner agent can be camera, dirt sensor and the actuator can be motor to make it move, absorption mechanism. And it can be represented as :

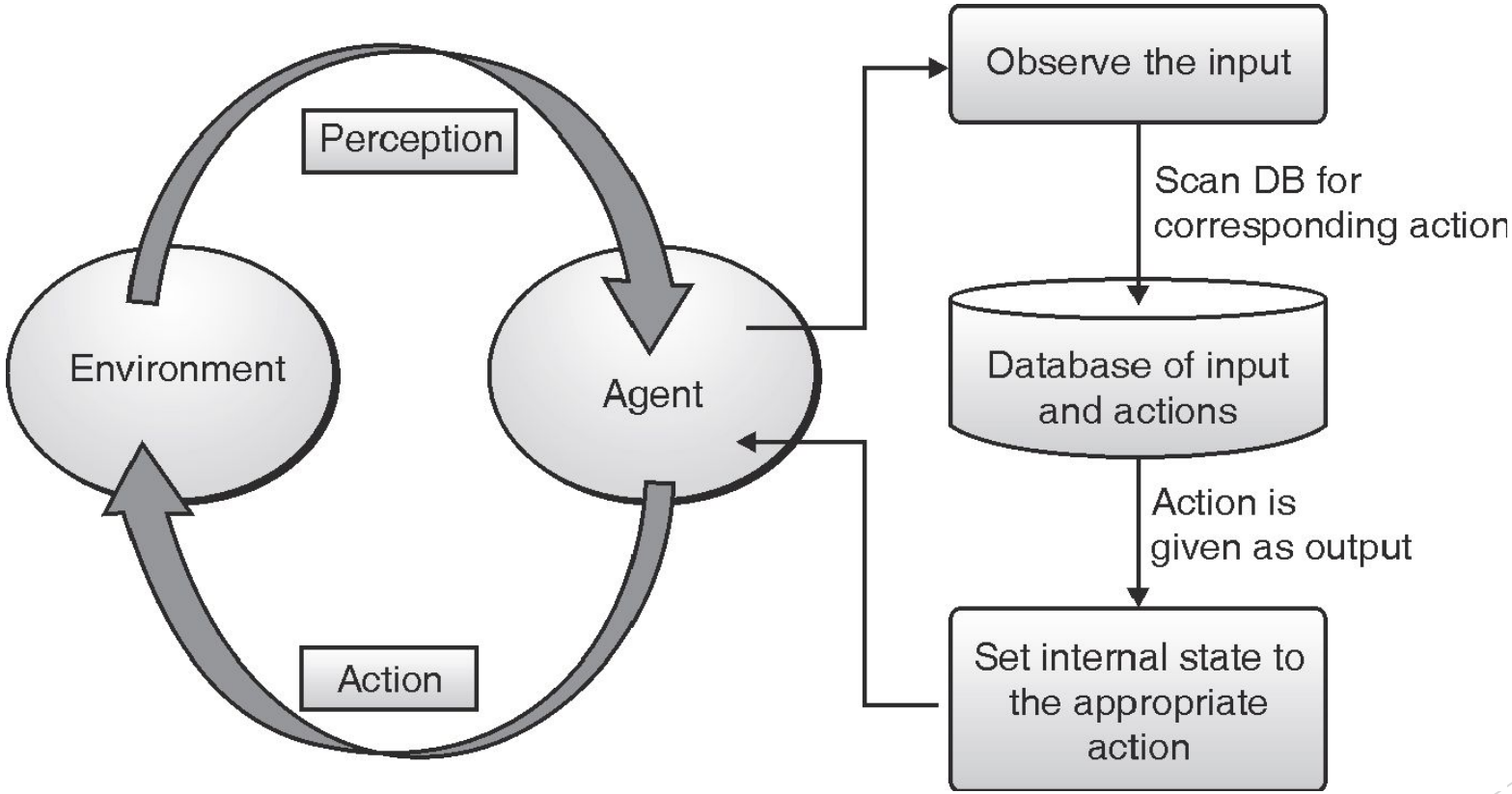
  [A, Dirty], [B, Clean], [A, absorb],[B, Nop], etc.

- Based on the percepts, actions will be performed. For example : Move left, Move right, absorb, No Operation.

Intelligent Agent

- **Intelligent agent** is the one which can take input from the environment through its sensors and act upon the environment through its actuators. Its actions are always directed to achieve a goal.
- In case of intelligent agents, the software modules are responsible for exhibiting intelligence. Generally observed capabilities of an intelligent agent can be given as follows:
- Ability to remain autonomous (Self-directed)
- Responsive
- Goal-Oriented

Structure of Intelligent Agents



Real life example

- Let’s understand this working with a real life example. Consider you are an agent and your surroundings is an environment. Now, take a situation where you are cooking in kitchen and by mistake you touch a hot pan.
- We will see what happens in this situation step by step. Your touch sensors take input from environment (i.e. you have touched some hot element), then it asks your brain if it knows “what action should be taken when you go near hot elements?”
- Now the brain will inform your hands (actuators) that you should immediately take it away from the hot element otherwise it will burn. Once this signal reaches your hand you will take your hand away from the hot pan.

flexible self-governed actions

- An intelligent agent is one that is capable of taking flexible self-governed actions”.
- Flexible means three things:
- 1. Reactiveness
- 2. Pro-activeness
- 3. Social Ability

Reactiveness

- It means giving reaction to a situation in a stipulated **time frame**. An agent can perceive the environment and respond to the situation in a particular time frame.
- In case of reactiveness, reaction within situation time frame is more important.
- You can understand this with above example, where, if an agent takes more time to take his hand away from the hot pan then agents hand will be burnt.

Pro-activeness

- It is **controlling a situation** rather than just responding to it. Intelligent agent show goal-directed behavior by taking the initiative.
- For example : If you are playing chess then winning the game is the main objective. So here we try to control a situation rather than just responding to one-one action which means that killing or losing any of the 16 pieces is not important, whether that action can be helpful to checkmate your opponent is more important.

Social ability

- Intelligent agents can interact with other agents (also humans). Take automatic car driver example
- where agent might have to interact with other agent or a human being while driving the car.

Few more features of an intelligent agent.

- **Self-Learning :** An intelligent agent changes its behavior based on its previous experience. This agent keeps updating its knowledge base all the time.
- **Movable/Mobile :** An Intelligent agent can move from one machine to another while performing actions.
- **Self-governing :** An Intelligent agent has control over its own actions.

Topics Covered in the last class

- Categorization of Intelligent Systems
- Components of AI- Perception, Knowledge Representation ,Learning , Reasoning, Problem Solving, NLP
- Applications of AI
- What is Agent?
- Intelligent Agent

Rational Agent

- For problem solving, if an agent makes a decision based on some logical reasoning, then, the decision is called as a “Rational Decision”
- A rational agent is an agent that has clear preferences, can model uncertainty via expected values of variables or functions of variables, and always chooses to perform the action with the optimal expected outcome for itself from among all feasible actions

Rationality depends on four main criteria

- **Performance measure** which defines the criterion of success for an agent
- Agent's **prior knowledge** of the environment
- **Action** performed by the agent
- Agent's **percept sequence** to date.

Performance measure

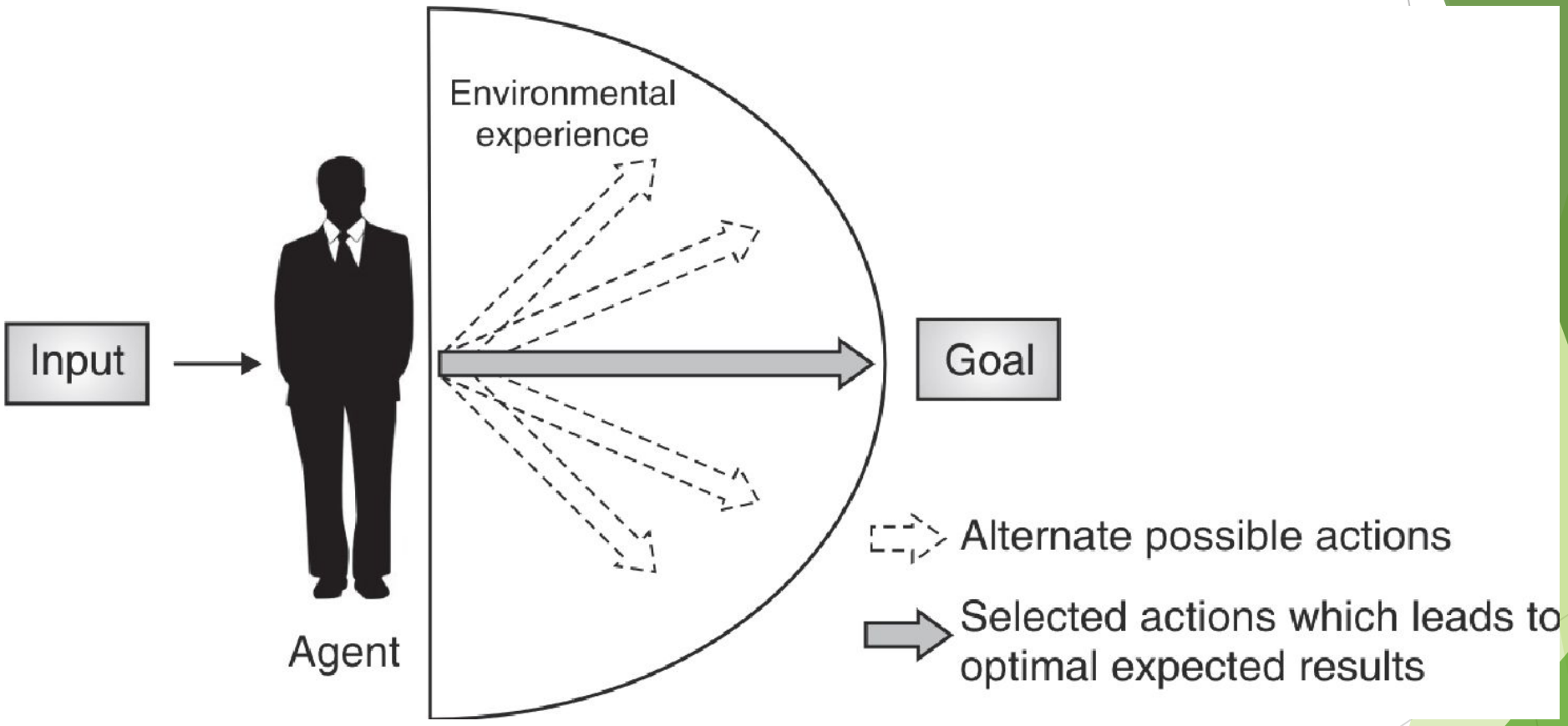
- For every percept sequence a built-in knowledge base is updated, which is very useful for decision making, because it stores the consequences of performing some particular action.
- If the consequences direct to achieve desired goal then we get a good performance measure factor, else, if the consequences do not lead to desired goal state, then we get a poor performance measure factor.

Example…

(a) Agent's finger is hurt while using nail and hammer

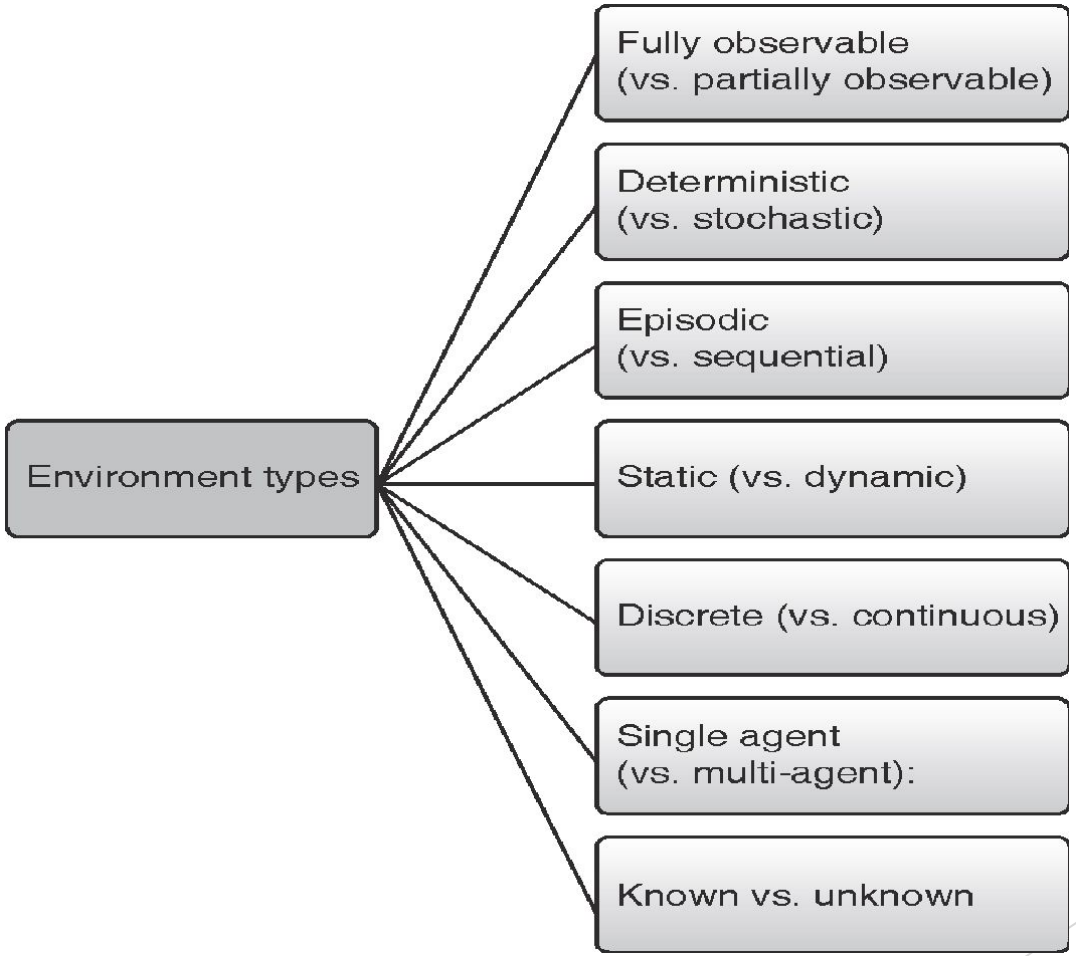
(b) Agent is using nail and hammer efficiently

Rational agent



Rational agent can be defined as an agent who makes use of its percept sequence, experience and knowledge to maximize the performance measure of an agent for every probable action. It selects the most feasible action which will lead to the expected results optimally.

Environments Types / Nature of Environment / Agent Environments



Fully observable vs. Partially observable

- The first type of environment is based on the observability. Whether the agent sensors can have access to complete state of environment at any given time or not, decides if it is a fully observable or partially observable environment.
- In **Fully observable** environments agents are able to gather all the necessary information required to take actions.
- Also in case of fully observable environments agents don't have to keep records of internal states.
- For example, Word-block problem, 8-puzzle problem, Sudoku puzzle, etc. in all these problem , the state is completely visible at any point of time.

Partially observable

- Environments are called **partially observable** when sensors cannot provide errorless information at any given time for every internal state, as the environment is not seen completely at any point of time.
- Also there can be unobservable environments where the agent sensors fail to provide information about internal states.
- For example, In case of an **automated car driver system**, automated car cannot predict what the other drivers are thinking while driving cars. Only because of the sensor's information gathering expertise it is possible for an automated car driver to take the actions.

Single agent vs. Multi-agent

- The second type of an environment is based on the number of agents acting in the environment. Whether the agent is operating on its own or in collaboration with other agents decides if it is a **Single agent** or a multi-agent environment.
- For example : An agent playing Tetris by itself can be a single agent environment, whereas we can have an agent playing checkers in a two-agent environment.
- Or in case of **vacuum cleaner world**, only one machine is working, so it's a **single agent** while in case of **car driving agent**
- There are multiple agents driving on the road, hence it's a **multi-agent environment**.

Co-operative multi-agent and Competitive multi-agent

- Now, you might be thinking in case of an automated car driver system which type of agent environment do we have?
- Let's understand it with the help of an automated car driving example. For a car driving system 'X', other car say 'Y' is considered as an Agent. When 'Y' tries to maximize its performance measure and the input taken by car ‘Y’ depends on the car 'X'. Thus it can be said that for an automated car driving system we have a **cooperative multi-agent environment.**
- Whereas in case of “chess game” when two agents are operating as opponents, and trying to maximize their own performance, they are acting in **competitive multi agent environment.**

Deterministic vs. Stochastic

- An environment is called **deterministic environment**, when the next state of the environment can be completely determined by the previous state and the action executed by the agent.
- For example, in case of vacuum cleaner world, 8-puzzle problem, chess game the next state of the environment solely depends on the current state and the action performed by agent.
- **Stochastic environment** generally means that the indecision about the actions is enumerated in terms of probabilities. That means environment changes while agent is taking action, hence the next state of the world does not merely depends on the current state and agent's action. And there are few changes happening in the environment irrespective of the agent's action.
- An automated car driving system has a stochastic environment as the agent cannot control the traffic conditions on the road.

Strategic

- If the environment is deterministic except for the actions of other agents, then the environment is **strategic.** That is, in case of game like chess, the next state of environment does not only depend upon the current action of agent but it is also influenced by the strategy developed by both the opponents for future moves.

Episodic vs. Sequential

- An **episodic task environment** is the one where each of the agent's action is divided into an atomic incidents or episodes. The current incident is different than the previous incident and there is no dependency between the current and the previous incident. In each incident the agent receives an input from environment and then performs a corresponding action.
- Generally, classification tasks are considered as episodic. Consider an example of pick and place robot agent, which is used to detect defective parts from the conveyor belt of an assembly line. Here, every time agent will make the decision based on current part, there will not be any dependency between the current and previous decision.

Sequential environments

- In **sequential environments**, as per the name suggests, the previous decision can affect all future decisions. The
- next action of the agent depends on what action he has taken previously and what action he is supposed to take in future.
- For example, in checkers where previous move can affect all the following moves. Also sequential environment can be understood with the help of an automatic car driving example where, current decision can affect the next decisions.
- If agent is initiating breaks, then he has to press clutch and lower down the gear as next consequent actions.

Static vs. Dynamic

- You have learnt about static and dynamic terms in previous semesters with respect to web pages. Same way we have **static (vs. dynamic) environments.** If an environment remains unchanged while the agent is performing given tasks then it is called as a static environment. For example, Sudoku puzzle or vacuum cleaner environment are static in nature.
- If environment is not changing over the time but, an agent's performance is changing then, it is called as **a semi-dynamic** environment. That means, there is a timer exist in the environment who is affecting the performance of the agent.

Static vs. Dynamic

- For example, In chess game or any puzzle like block word problem or 8-puzzle if we introduce timer, and if agent's performance is calculated by time taken to play the move or to solve the puzzle, then it is called as semi-dynamic environment.
- Lastly, if the environment changes while an agent is performing some task, then it is called **dynamic environment**.
- In this type of environment agent's sensors have to continuously keep sending signals to agent about the current state of the environment so that appropriate action can be taken with immediate effect.
- Automatic car driver example comes under dynamic environment as the environment keeps changing all the time.

Discrete vs. Continuous

- You have seen discrete and continuous signals in old semesters. When you have distinct, quantized, clearly defined values of a signal it is considered as discrete signal.
- Same way, when there are distinct and clearly defined inputs and outputs or precepts and actions, then it is called a **discrete environment**.
- For example : chess environment has a finite number of distinct inputs and actions.
- When a continuous input signal is received by an agent, all the precepts and actions cannot be defined beforehand then it is called **continuous environment**. For example : An automatic car driving system.

Known vs. Unknown

- In a **known environment**, the output for all probable actions is given.
- in case of **unknown environment,** for an agent to make a decision, it has to gain knowledge about - how the environment works.

Examples….

Task environment	Car driving	Part – Picking Robot	Cross word puzzle	Soccer game	Checkers with clock
Observable	Partially	Partially	fully	Partially	Fully
Agents	Multi agent (cooperative)	Single agent	single	Multi agent (competitive)	Multi agent (competitive)
Deterministic	Stochastic	Stochastic	Deterministic	Strategic	Strategic

Examples….

Task environment	Car driving	Part – Picking Robot	Cross word puzzle	Soccer game	Checkers with clock
Episodic	Sequential	Episodic	Sequential	Sequential	Sequential
Static	Dynamic	Dynamic	Static	Dynamic	Semi
Discrete	Continuous	Discrete	Discrete	Continuous	Discrete
Known and Unknown	Unknown	Known	Known	Known	Known

Quiz Time

Identify Environment, Sensors and Actuators in the following agents

Medical Diagnosis System

Interactive English Tutor

Refinery Controller

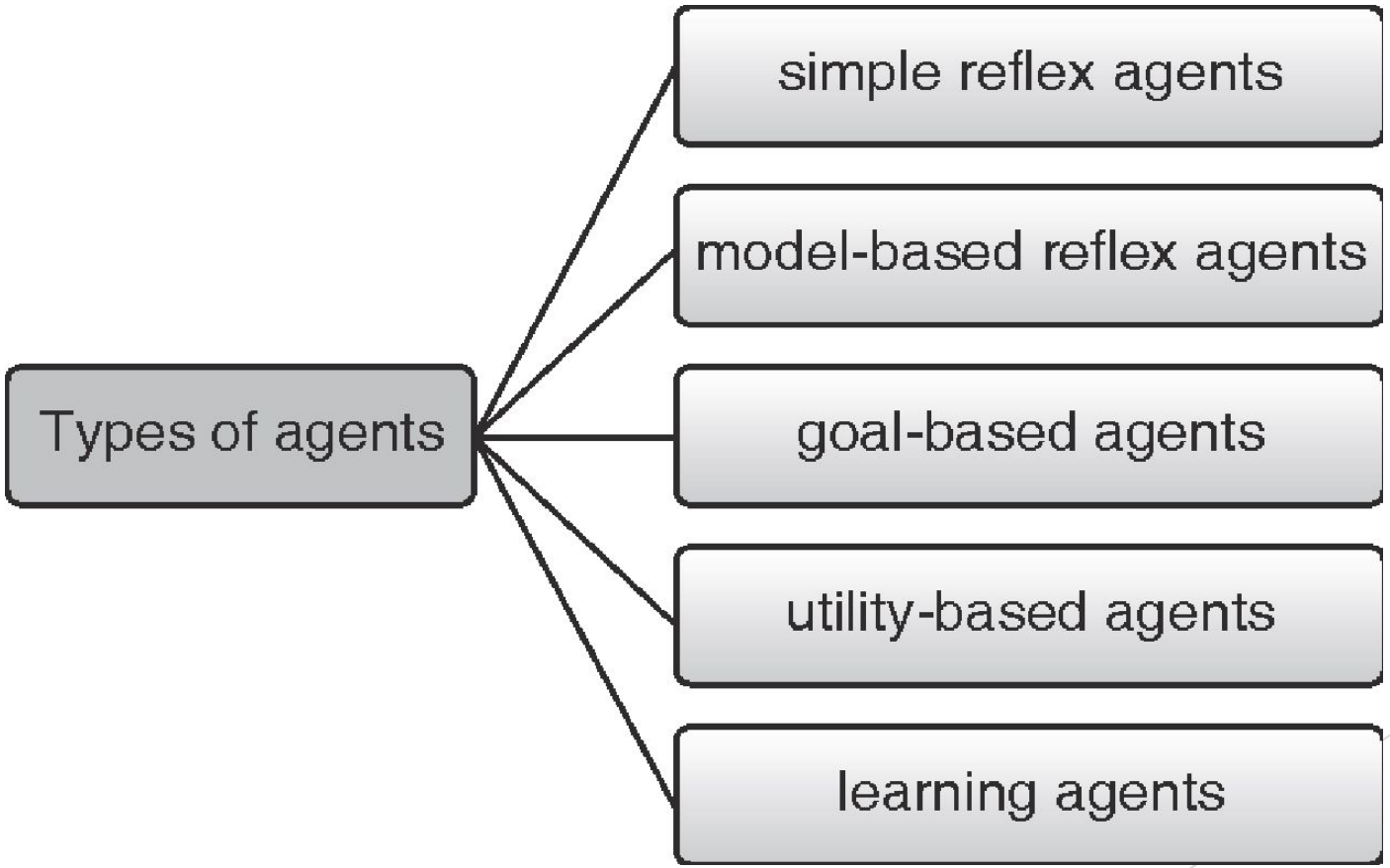
Auction Bidding

AI Book Shopping

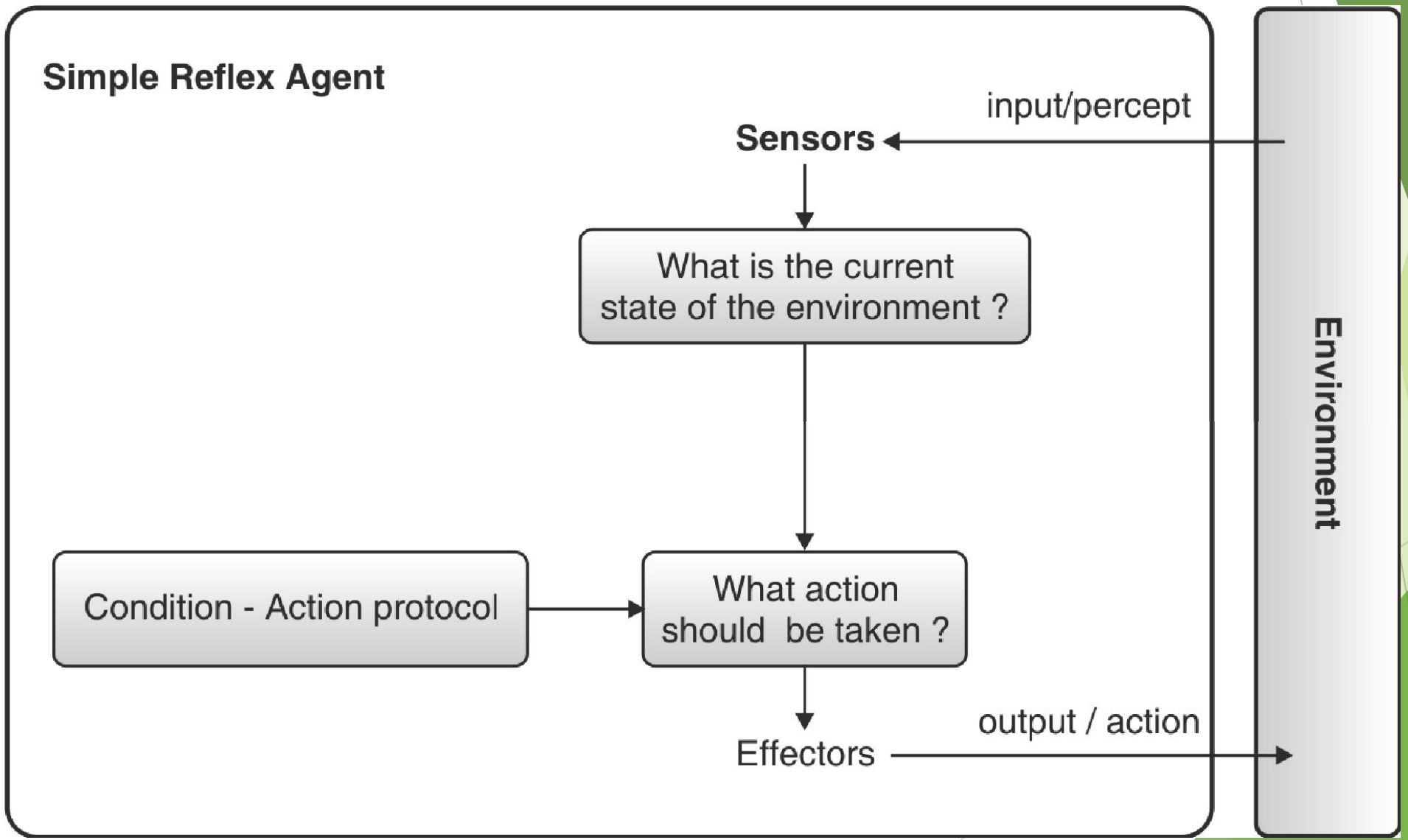
Topics covered in the last class

- Rational Agent
- Fully observable/partially observable
- Multi-agent/Single agent environment
- Deterministic/stochastic
- Episodic environment/Sequential environment
- Dynamic environment/static environment
- Discrete /continuous environments
- Known and unknown environments

Types of Agents



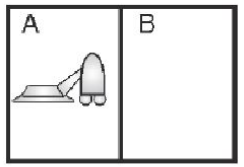
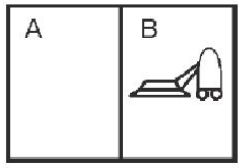
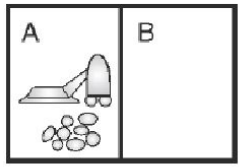
Simple Reflex Agents



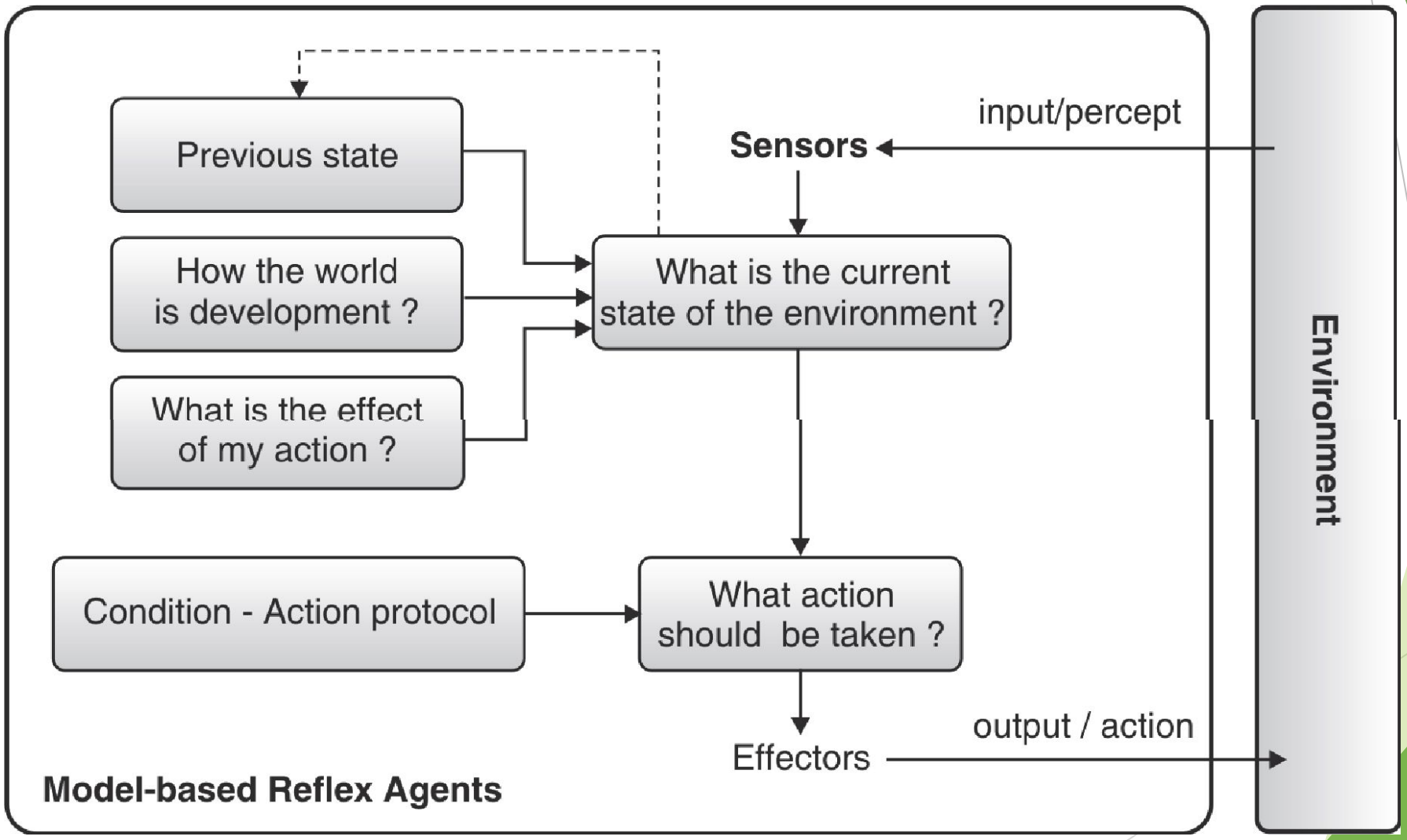
Simple Reflex Agents

- You can understand simple reflexes with the help of a real life example, say some object approaches eye then, you will blink your eye. This type of simple reflex is called natural/innate reflex.
- Consider the example of the vacuum cleaner agent. It is a simple reflex agent, as its decision is based only on whether the current location contains dirt.

Few possible input sequences and outputs for vacuum cleaner world with 2 locations are considered for simplicity.

Figure	Input sequence {location, content }	Output / action Right, left, suck, no-op
	{A, clean}	Right
	{B, clean}	Left
	{A, dirt}	Suck

Model-Based Reflex Agents



Model-Based Reflex Agents

- An agent which performs actions based on the current input and one previous input is called as model-based agent.
- Partially observable environment can be handled well by model-based agent.
- once the sensor takes input from the environment, agent checks for the current state of the environment.
- After that, it checks for the previous state which shows how the world is developing and how the environment is affected by the action which was taken by the agent at earlier stage. This is termed as model of the world.
- Once this is verified, based on the condition-action protocol an action is decided. This decision is given to effectors and the effectors give this output to the environment

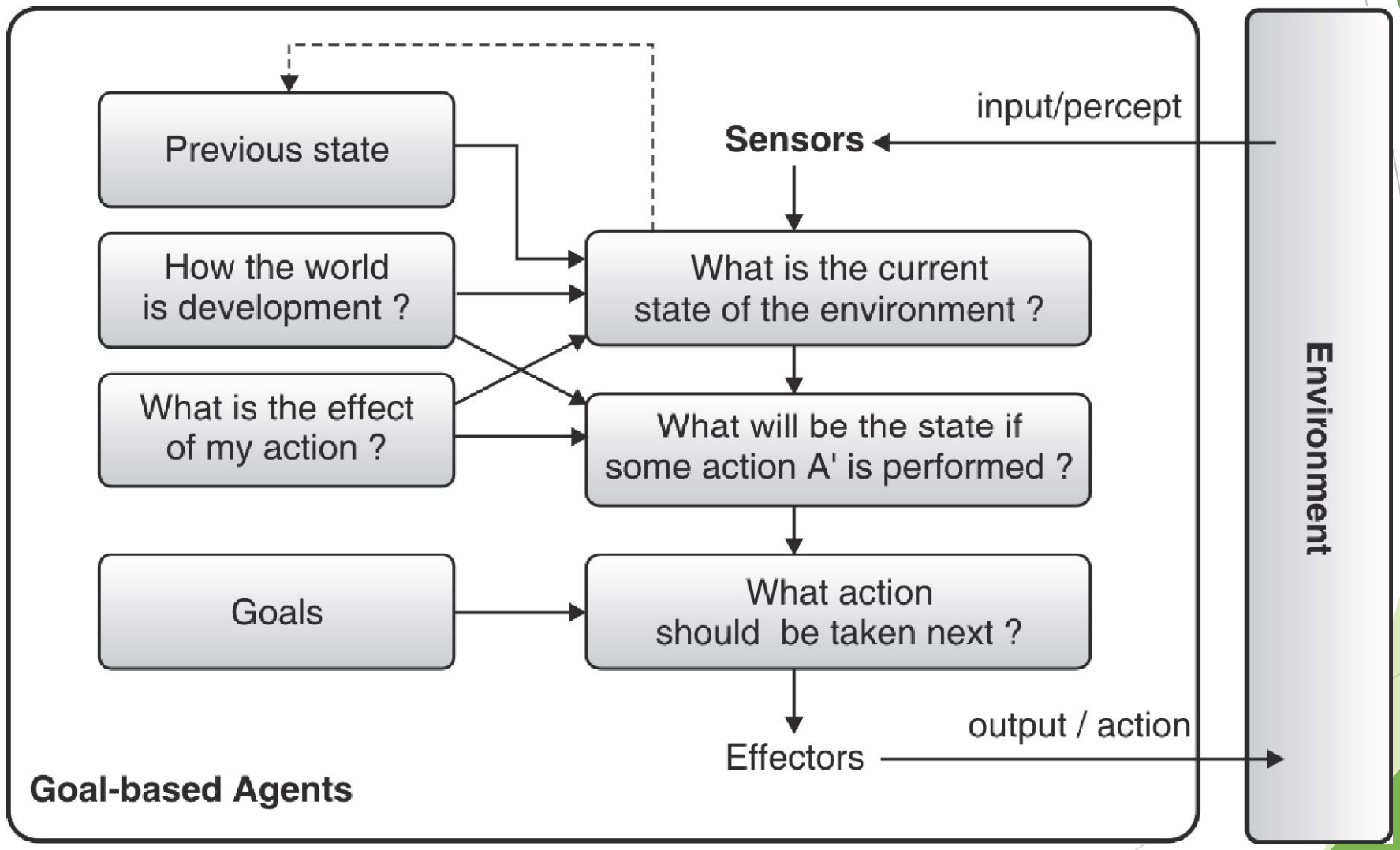
Model-Based Reflex Agents

- Consider a simple example of automated car driver system. Here, the world keeps changing all the time. You must have taken a wrong turn while driving on some or the other day of your life. Same thing applies for an agent.
- Suppose if some car “X” is overtaking our automated driver agent “A”, then speed and direction in which “X” and “A” are moving their steering wheels is important. Take a scenario where agent missed a sign board as it was overtaking other car. The world around that agent will be different in that case.

Model-Based Reflex Agents

- Internal model based on the input history should be maintained by model-based reflex agent, which can reflect at least some of the unobserved aspects of the current state.
- Once this is done it chooses an action in the same way as the simple reflex agent

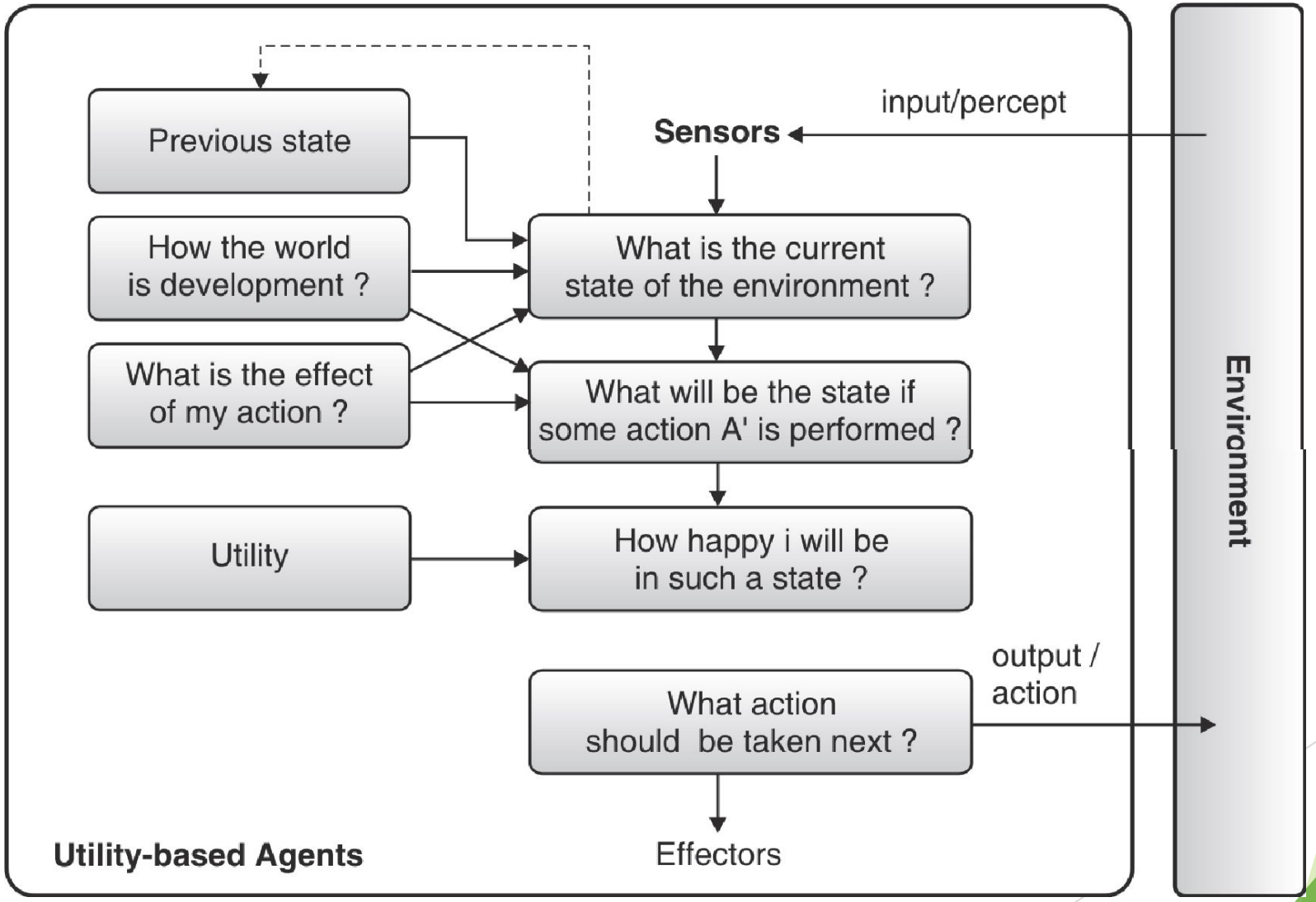
Goal-Based Agents



Quiz Time

- Important vocabulary:
- perception
- action
- sensor
- actuator
- environment
- performance metric
- Deterministic/stochastic
- fully observable/partially observable
- dynamic environment/static environment
- multi-agent/Single agent environment
- known and unknown environments
- Episodic environment/Sequential environment
- discrete vs. continuous environments

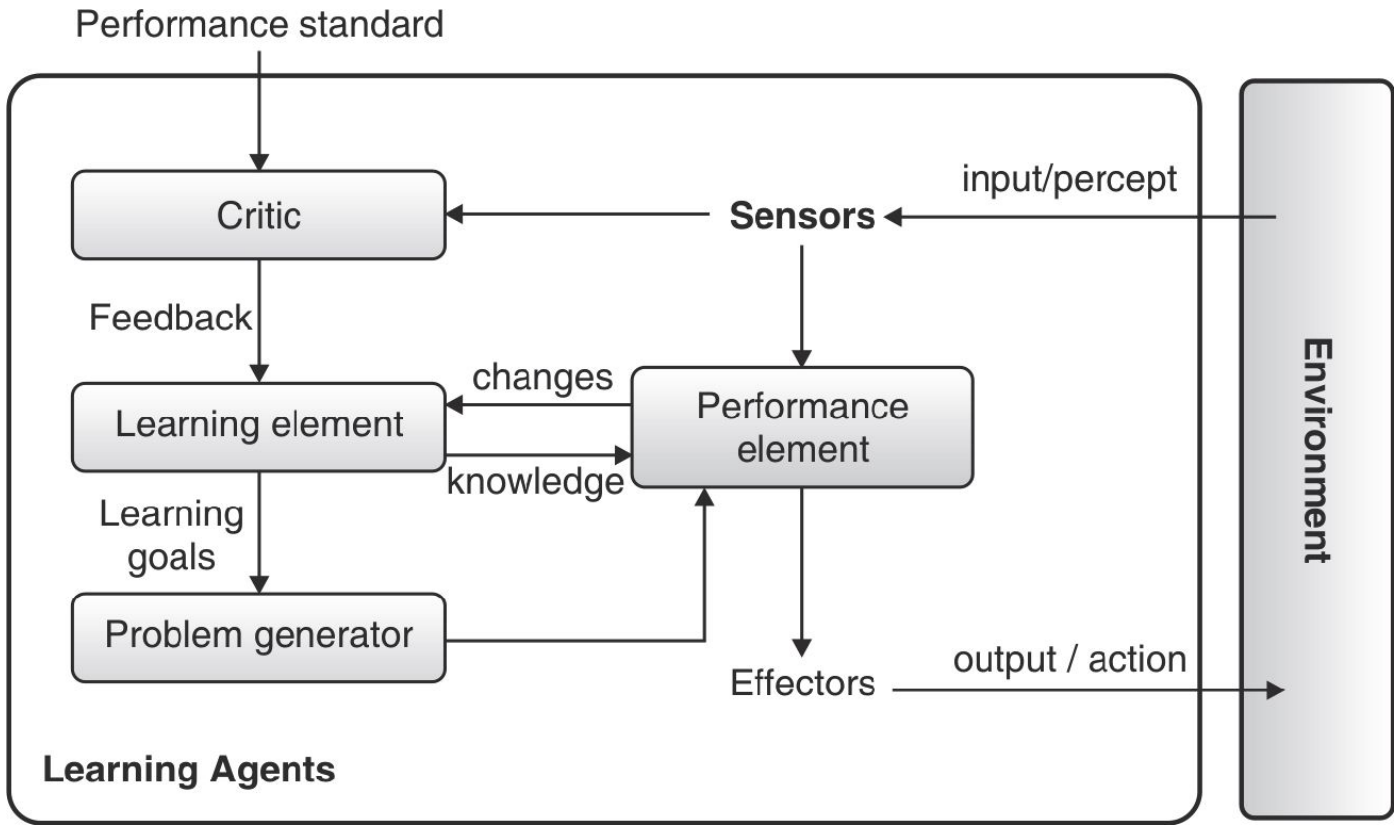
Utility-Based Agents



Utility-Based Agents

- Take one example; you might have used Google maps to find out a route which can take you from source location to your destination location in least possible time.
- Same logic is followed by utility based automatic car driving agent.
- ˜ Goals utility based automatic car driving agent can be used to reach given location safely within least possible time and save fuel.
- So this car driving agent will check the possible routes and the traffic conditions on these routes and will select the route which can take the car at destination in least possible time safely and without consuming much fuel.

Learning Agents



Components of learning agent

1. Critic
2. Learning element
3. Performance element
4. Problem generator

Components of learning agent

1. Critic : It is the one who compares sensor's input specifying effect of agent's action on the environment with the performance standards and generate feedback for leaning element.

2. Learning element : This component is responsible to learn from the difference between performance standards and the feedback from critic. According to the current percept it is supposed to understand the expected behavior and enhance its standards

3. Performance element : Based on the current percept received from sensors and the input obtained by the learning element, performance element is responsible to choose the action to act upon the external environment.

4. Problem generator : Based on the new goals learnt by learning agent, problem generator suggests new or alternate actions which will lead to new and instructive understanding.

Topics covered in the last class

- **Types of Agents**
  - Simple Reflex Agents
  - Model-Based Reflex Agents
  - Goal-Based Agents
  - Utility-Based Agents
  - Learning Agent

PEAS representation for an agent

- **PEAS** : PEAS stands for **Performance Measure, Environment, Actuators, and Sensors.**
- It is the short form used for performance issues grouped under Task Environment.

PEAS

- **Performance Measure :** It the objective function to judge the performance of the agent. For example, in case of pick and place robot, number of correct parts in a bin can be the performance measure.
- **Environment :** It the real environment where the agent need to deliberate actions.
- **Actuators :** These are the tools, equipment or organs using which agent performs actions in the environment. This works as the output of the agent.
- **Sensors :** These are the tools, equipment or organs using which agent captures the state of the environment. This works as the input to the agent.

(A) Automated car driving agent

- 1. Performance measures which should be satisfied by the automated car driver:
- **(i) Safety :** Automated system should be able to drive the car safely without dashing anywhere.
- **(ii) Optimum speed :** Automated system should be able to maintain the optimal speed depending upon the surroundings.
- **(iii) Comfortable journey :** Automated system should be able to give a comfortable journey to the end user, i.e. depending upon the road it should ensure the comfort of the end user.
- **(iv) Maximize profits :** Automated system should provide good mileage on various roads, the amount of energy consumed to automate the system should not be very high, etc. such features ensure that the user is benefited with the automated features of the system and it can be useful for maximizing the profits

2. Environment

- **(i) Roads :** Automated car driver should be able to drive on any kind of a road ranging from city roads to highway.
- **(ii) Traffic conditions :** You will find different set of traffic conditions for different type of roads. Automated system should be able to drive efficiently in all types of traffic conditions. Sometimes traffic conditions are formed because of pedestrians, animals, etc.
- **(iii) Clients :** Automated cars are created depending on the client's environment. For example, in some countries you will see left hand drive and in some countries there is a right hand drive. Every country/state can have different weather conditions. Depending upon such constraints automated car driver should be designed.

3. Actuators

Actuators are responsible for performing actions/providing output to an environment.

- In case of car driving agent following are the actuators :
- **(i) Steering wheel** which can be used to direct car in desired direction (i.e. right/left)
- **(ii) Accelerator, gear**, etc. can be useful to increase or decrease the speed of the car.
- **(iii) Brake** is used to stop the car.
- **(iv) Light signal, horn** can be very useful as indicators for an automated car.

4. Sensors

- To take input from environment in car driving example cameras, sonar system, speedometer, GPS, engine sensors, etc. are used as sensors.

(B) Part-picking ARM robot

- **(i) Performance measures** : Number of parts in correct container.
- **(ii) Environment :** Conveyor belt used for handling parts, containers used to keep parts.
- **(iii) Actuators** : Arm with tooltips, to pick and drop parts from one place to another.
- **(iv) Sensors** : Camera to scan the position from where part should be picked and joint angle sensors which are used to sense the obstacles and move in appropriate place.

(C) Medical diagnosis system

- **(i) Performance measures**
- a. **Healthy patient:** system should make use of sterilized instruments to ensure the safety (healthiness) of the patient.
- b. **Minimize costs :** The automated system results should not be very costly otherwise overall expenses of the patient may increase. Medical diagnosis system should be legal.
- **(ii) Environment :** Patient, Doctors, Hospital Environment
- **(iii) Sensors :** Screen, printer
- **(iv) Actuators :** Keyboard and mouse which is useful to make entry of symptoms, findings, patient's answers to given questions. Scanner to scan the reports, camera to click pictures of patients.

(D) Soccer player robot

- **(i) Performance measures :** Number of goals, speed, legal game.
- **(ii) Environment:** Team players, opponent team players, playing ground, goal net.
- **(iii) Sensors:** Camera, proximity sensors, infrared sensors.
- **(iv) Actuators :** Joint angles, motors.

Quiz Time !!! PEAS Examples

- AI Book Shopping
- Playing Tennis
- Auction Bidding
- Knitting Sweater

Topics covered in the last class

PEAS Representation

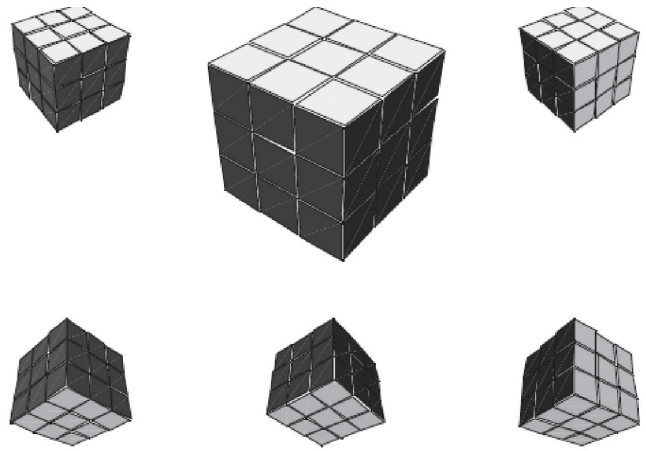
- **Performance Measure :** It the objective function to judge the performance of the agent. For example, in case of pick and place robot, number of correct parts in a bin can be the performance measure.
- **Environment :** It the real environment where the agent need to deliberate actions.
- **Actuators :** These are the tools, equipment or organs using which agent performs actions in the environment. This works as the output of the agent.
- **Sensors :** These are the tools, equipment or organs using which agent captures the state of the environment. This works as the input to the agent.

Problem Solving with Artificial Intelligence

- Early work in the field of AI focused on formal tasks, like game playing, proving theorems, etc.
- Games like chess, checkers received good deal of attention, because in case of machine opponents, system was using the experience gained in previous games to improve the moves in next game.
- **Knowledge base** was scaled up for the problems which needed perception through vision and/or speech, natural language understanding, problem solving for medical diagnosis, chemical analysis, etc.

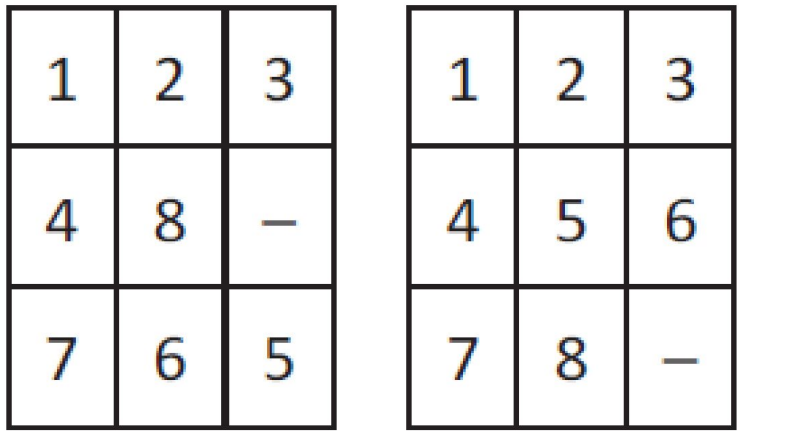
3*3*3 Rubik’s cube problem

- In Rubik’s cube, we have a cube with six color faces. The goal is to arrange all the cuboids in such a way that each face of cube will show distinct color as

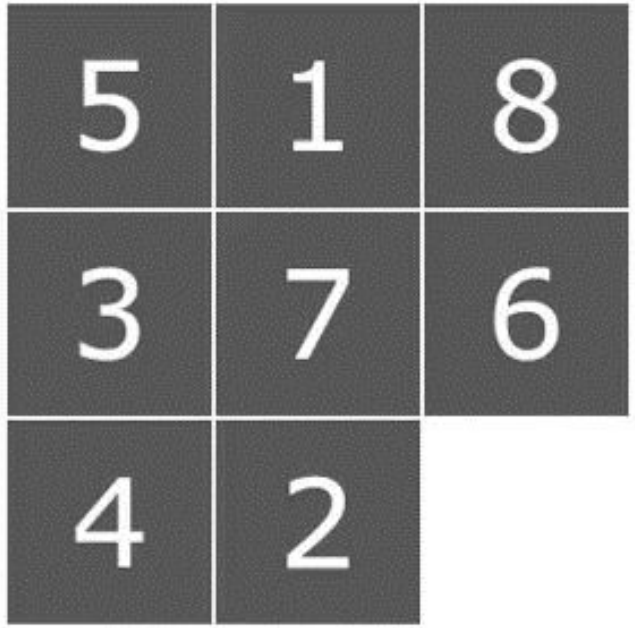


Example of 8-Puzzle Problem

- It has a 3 X 3 board with tiles having 1 through 8 numbers on it. There is a blank tile which can be moved forward, backward, to left and to right. The aim is to arrange all the tiles in the goal state form by moving the blank tile minimum number of times.

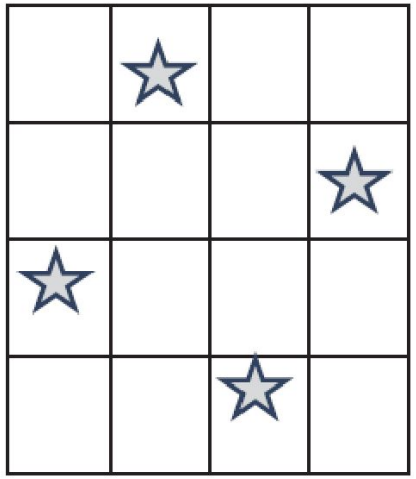


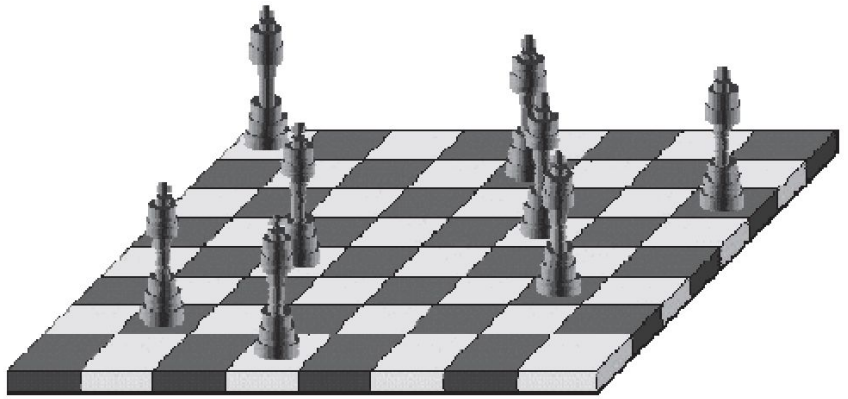
Example of 8-Puzzle Problem



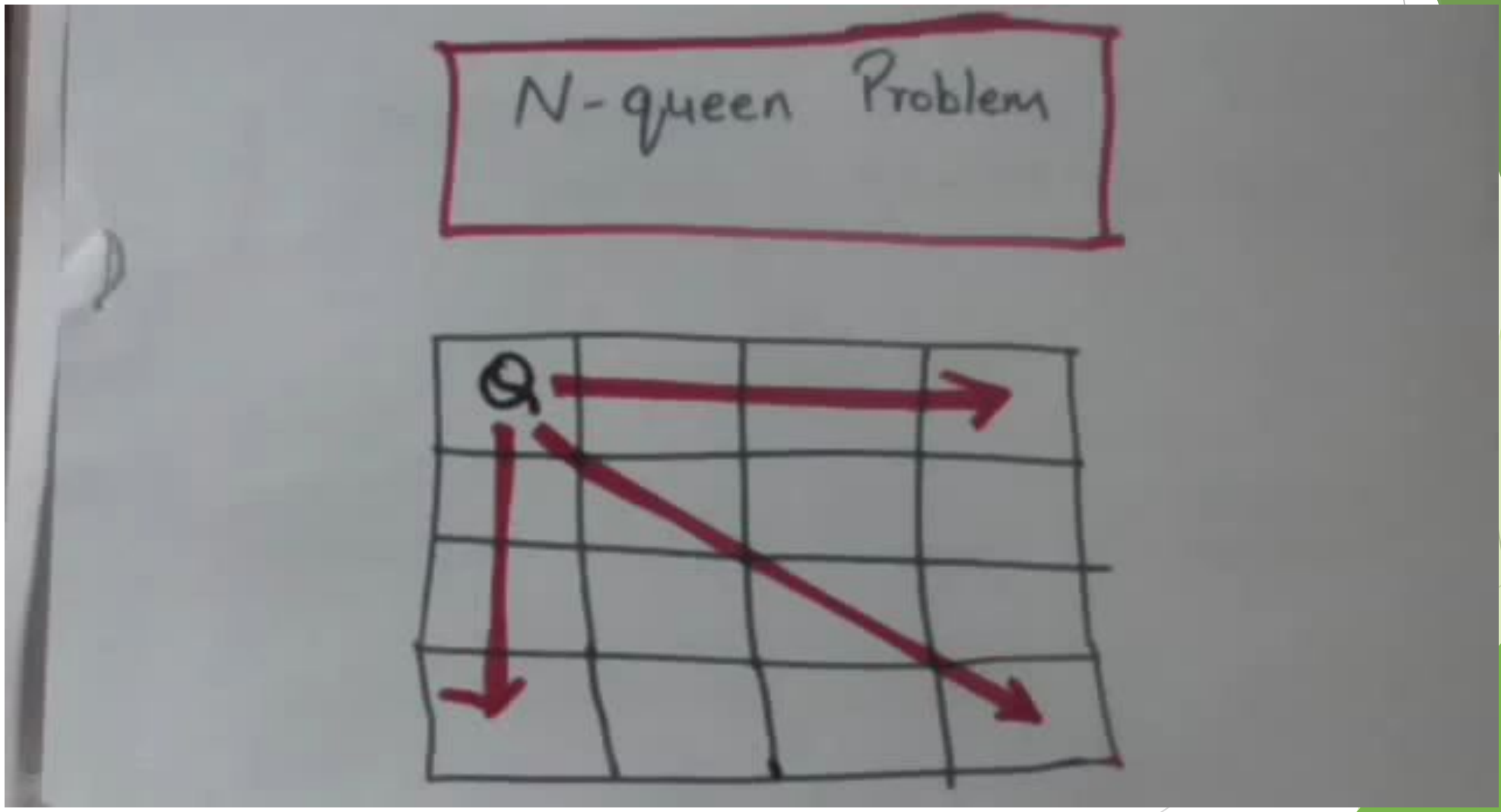
N-queen problem :

In n-queen, the queens need to be placed on the n*n board, in such a way that no queen can dash the other queen, horizontally, vertically or diagonally.





N-queen problem

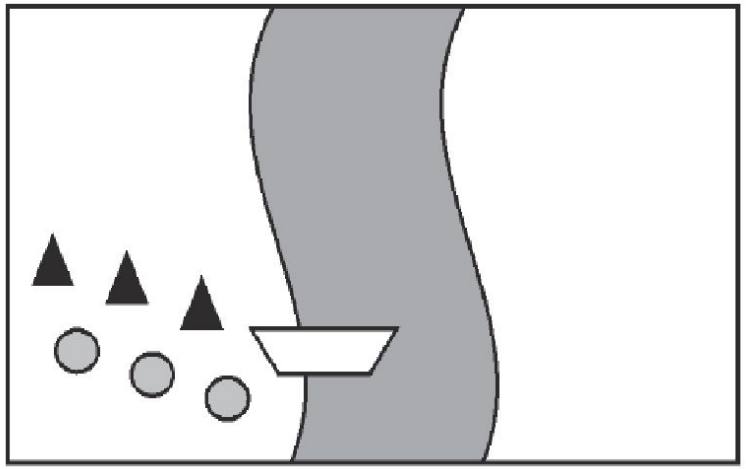


Missionaries and cannibals problem

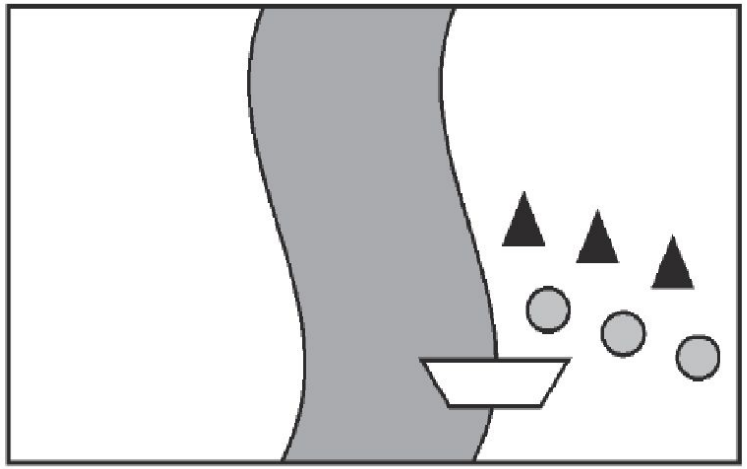
In this problem, there are three missionaries and three cannibals on the same side of a river.

− We need to get all of them to the other side of river through a canoe which can hold maximum two people at a time.

− The condition is no time during the process of shifting, number of cannibals on any of the side should be greater than the number of missionaries on the same side.



Initial Sate

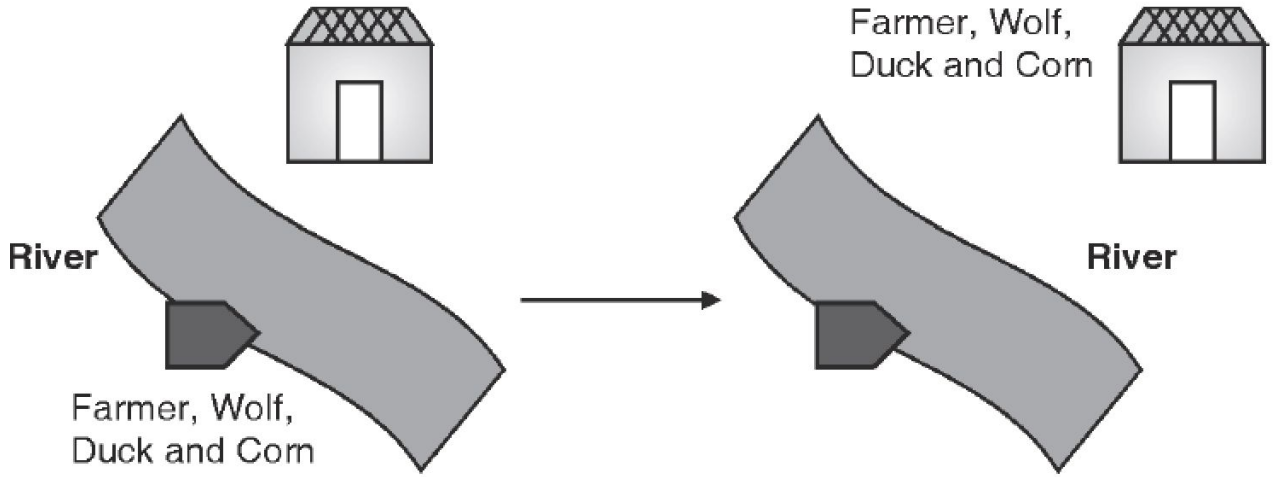


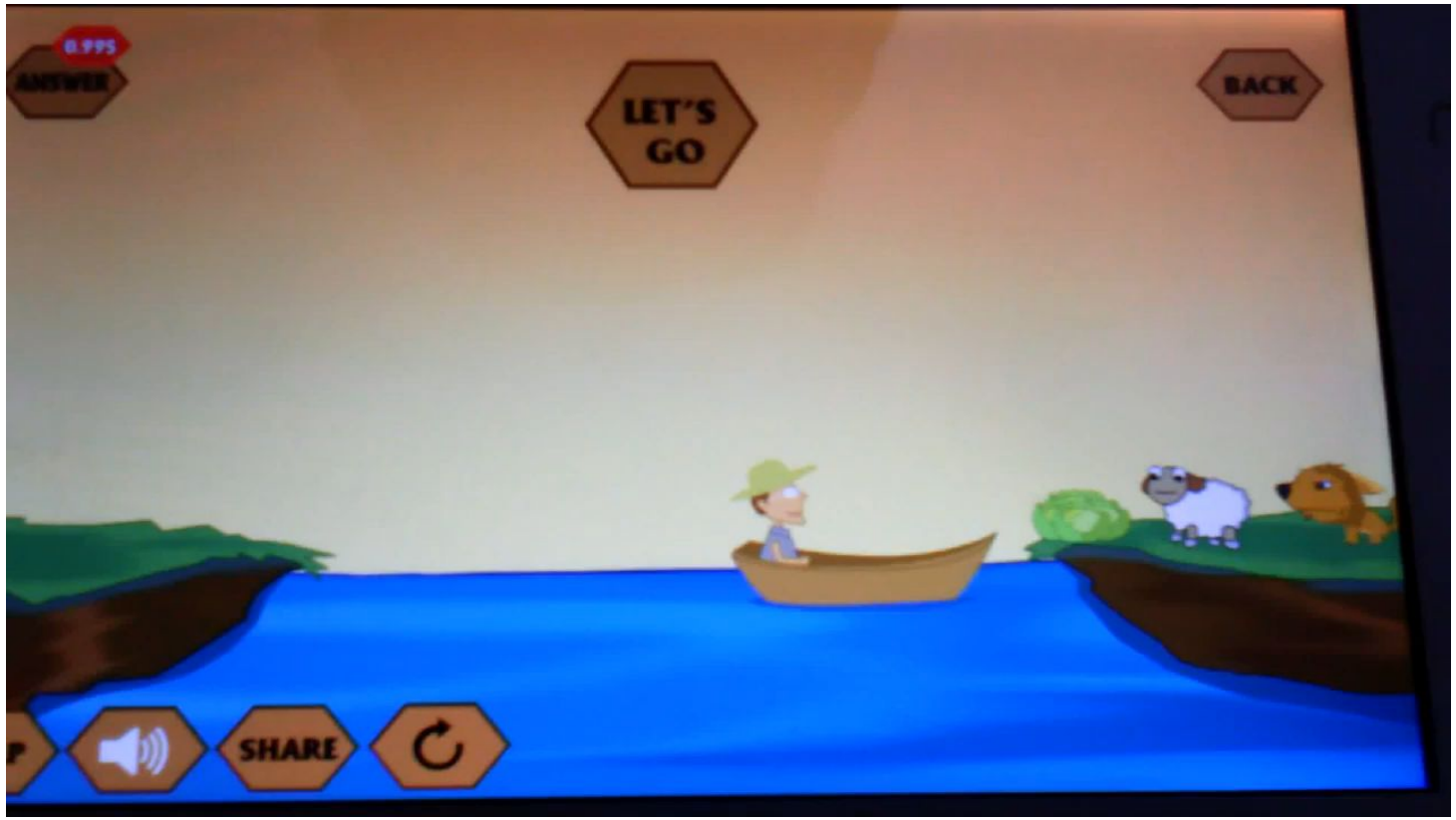
Final State

Missionaries and cannibals problem

The river problem :

- In this problem, a farmer needs to carry a wolf, a duck and corn across a river. The farmer has a small rowing boat, which can carry at most the farmer and one other thing.
- The problem is that, the wolf will eat the duck and the duck will eat the corn, if they are at the same side. How can the farmer safely transport the wolf, the duck and the corn to the opposite shore?




ANSWER
LET'S GO
BACK
SHARE

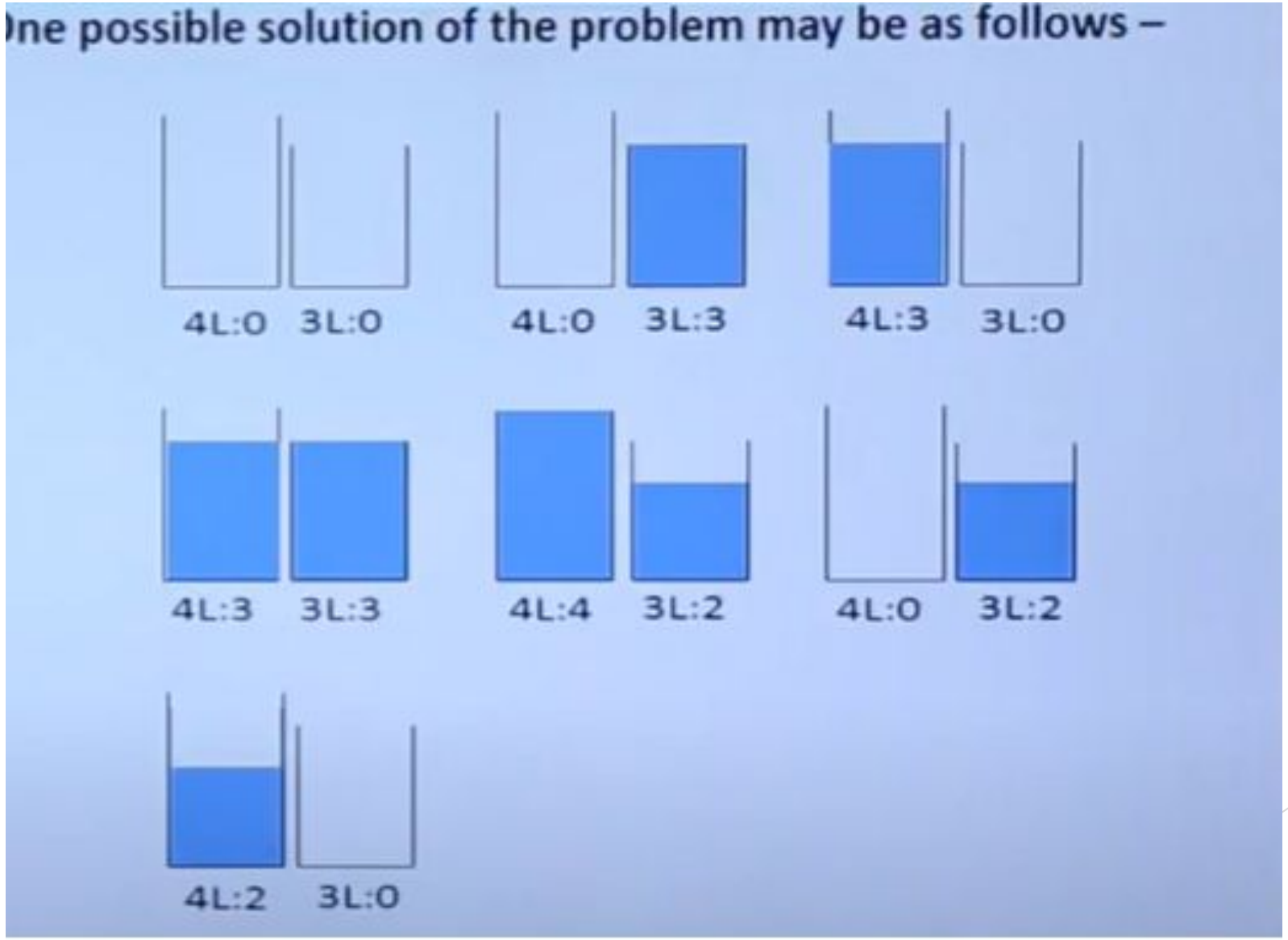
Water Jug Problem

Water Jugs Problem

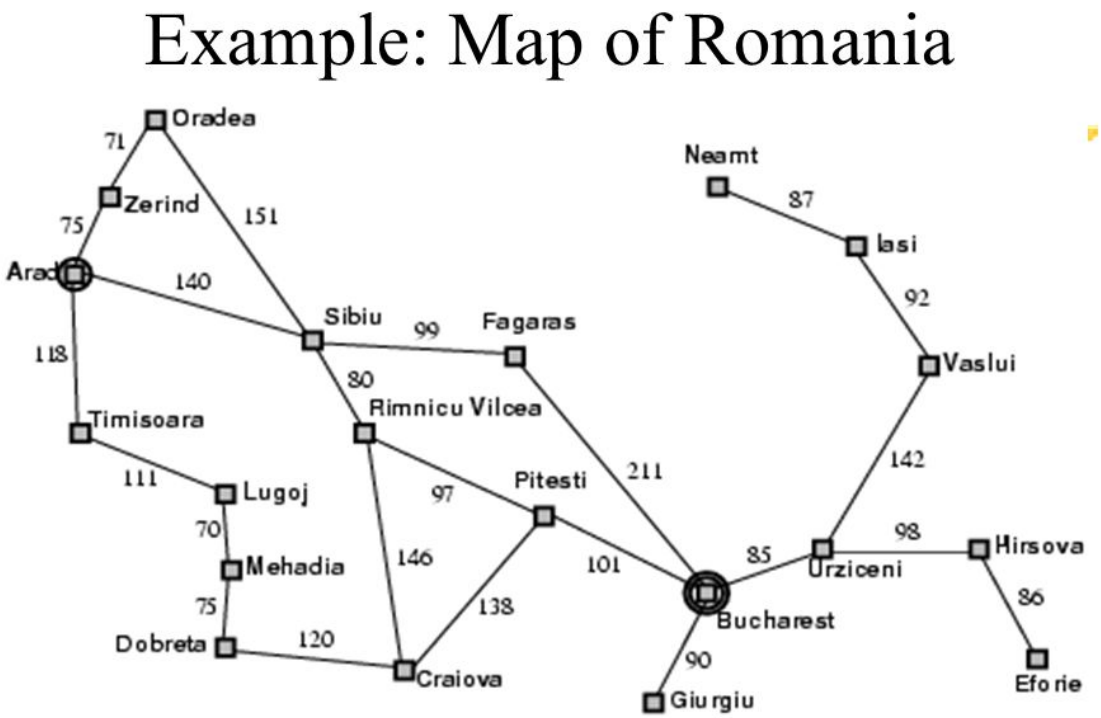
Given 4-liter and 3-liter pitchers, how do you get exactly 2 liters into the 4-liter pitcher?

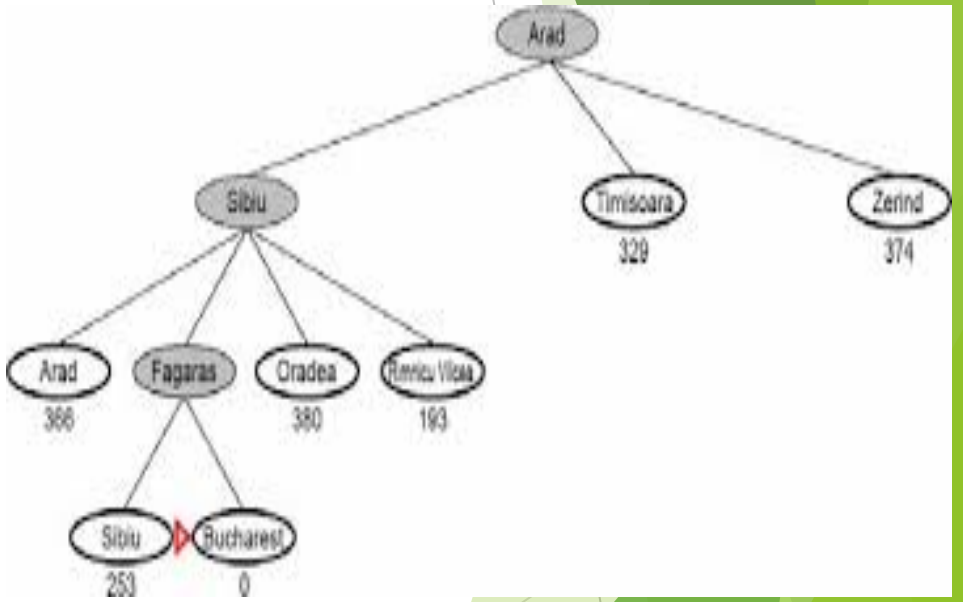
State: (*x*, *y*) for # liters in 4-liter and 3-liter pitchers, respectively
Actions: empty, fill, pour water between pitchers
Initial state: (0, 0)
Goal state: (2, *)

Water Jug Problem



Holiday in Romania

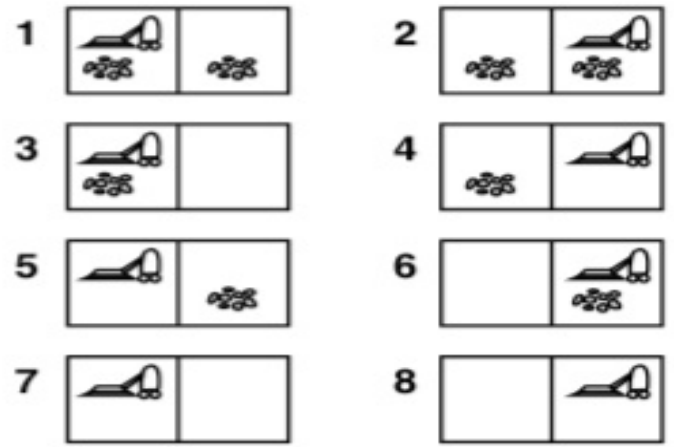




Vacuum World Problem

Example: vacuum world

- Single state, start in #5. Solution??



AI problems

- From the above examples, it must be clear that AI problems are the one , in which there are few conditions specified and the aim is to not only generate the solution but **also to improve performance of the system**;
- because that is where the intelligence of the system gets challenged. The term intelligence includes many cognitive skills, like the ability to

solve problems, learn, interpret and understand language.

Solving Problems by Searching

Now let us see how searching play a vital role in solving AI problems.

- Searching techniques can be used in game playing like **Tic-Tac-Toe** or navigation problems like **Travelling Salesman Problem**.
- ˜ First, we will understand the representation of given problem so that appropriate searching techniques can be applied to solve the problem.

Problem Formulation

- Given a goal to achieve; problem formulation is the process of **deciding what states to be considered** and what actions to be taken to achieve the goal. This is the first step to be taken by any problem solving agent.
- **State Space Representation :** The state space of a problem is the set of all states reachable from the initial state by executing any sequence of actions. State is representation of all possible outcomes.
- The state space specifies the relation among various problem states thereby, forming a graph in which the nodes are states and the links between nodes represent actions

Problem Formulation

- **State Space Search:** Searching in a given space of states pertaining to a problem under consideration is called a state space search.
- **Path :** A path is a sequence of states connected by a sequence of actions, in a given state space.

Well-Defined Problems and Solutions

Problem can be defined formally using five components as follows :

- 1. Initial state
- 2. Actions
- 3. Successor function
- 4. Goal test
- 5. Path cost

Well-Defined Problems and Solutions

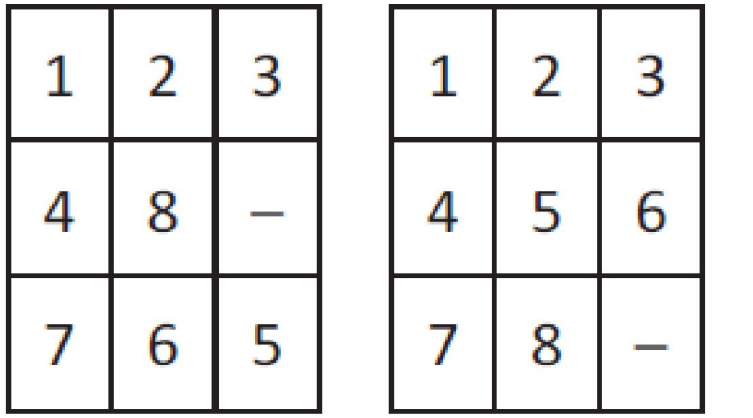
- **1. Initial state :** The initial state is the one in which the agent starts in.
- **2. Actions :** It is the set of actions that can be executed or applicable in all possible states. A description of what each action does; the formal name for this is the transition model.
- **3. Successor function :** It is a function that returns a state on executing an action on the current state.
- **4. Goal test :** It is a test to determine whether the current state is a goal state. In some problems the goal test can be carried out just by comparing current state with the defined goal state, called as **explicit goal test**. Whereas, in some of the problems, state cannot be defined explicitly but needs to be generated by carrying out some computations, it is called as **implicit goal test**.
- **For example :** In Tic-Tac-Toe game making diagonal or vertical or horizontal combination declares the winning state which can be compared explicitly; but in the case of chess game, the goal state cannot be predefined but it's a scenario called as **"Checkmate"**, which has to be evaluated implicitly.

Well-Defined Problems and Solutions

- **Path cost :** It is simply the cost associated with each step to be taken to reach to the goal state. To determine the cost to reach to each state, there is a cost function, which is chosen by the problem solving agent.
- **Problem solution :** A well-defined problem with specification of initial state, goal test, successor function, and path cost. It can be represented as a data structure and used to implement a program which can search for the goal state.
- **A solution** to a problem is a sequence of actions chosen by the problem solving agent that leads from the initial state to a goal state. Solution quality is measured by the path cost function.
- **Optimal solution :** An optimal solution is the solution with least path cost among all solutions.

Example of 8-Puzzle Problem

- A typical scenario of 8-puzzle problem. It has a 3 X 3 board with tiles having 1 through 8 numbers on it. There is a blank tile which can be moved forward, backward, to left and to right. The aim is to arrange all the tiles in the goal state form by moving the blank tile minimum number of times.



Example of 8-Puzzle Problem

- This problem can be formulated as follows :
- **States :** States can represented by a 3 x 3 matrix data structure with blank denoted by 0.
- **1. Initial state :** {{1, 2, 3},{4, 8, 0},{7, 6, 5}}
- **2. Actions :** The blank space can move in Left, Right, Up and Down directions specifying the actions.
- **3. Successor function :** If we apply “Down” operator to the start state in the resulting state has the 5 and the blank switched.
- **4. Goal test :** {{1, 2, 3},{4, 5, 6},{7, 8, 0}}
- **5. Path cost :** Number of steps to reach to the final state.

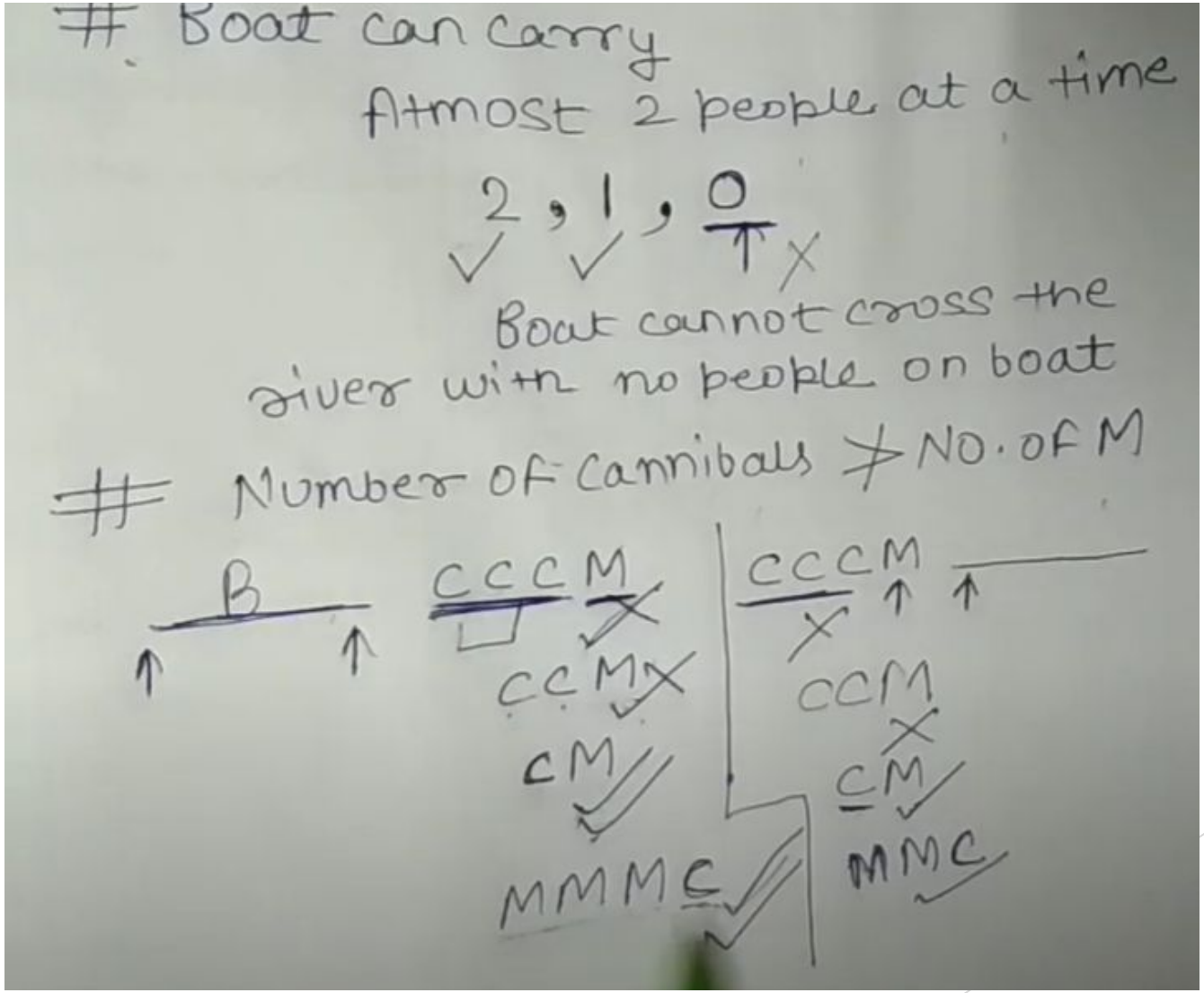
Example of 8-Puzzle Problem

- **Solution :**
- {{1, 2, 3}, {4, 8, 0}, {7, 6, 5}}
- {{1, 2, 3}, {4, 8, 5}, {7, 6, 0}}
- {{1, 2, 3}, {4, 8, 5}, {7, 0, 6}}
- {{1, 2, 3}, {4, 0, 5}, {7, 8, 6}}
- {{1, 2, 3}, {4, 5, 0}, {7, 8, 6}}
- {{1, 2, 3}, {4, 5, 6}, {7, 8, 0}}
- Path cost = 5 steps

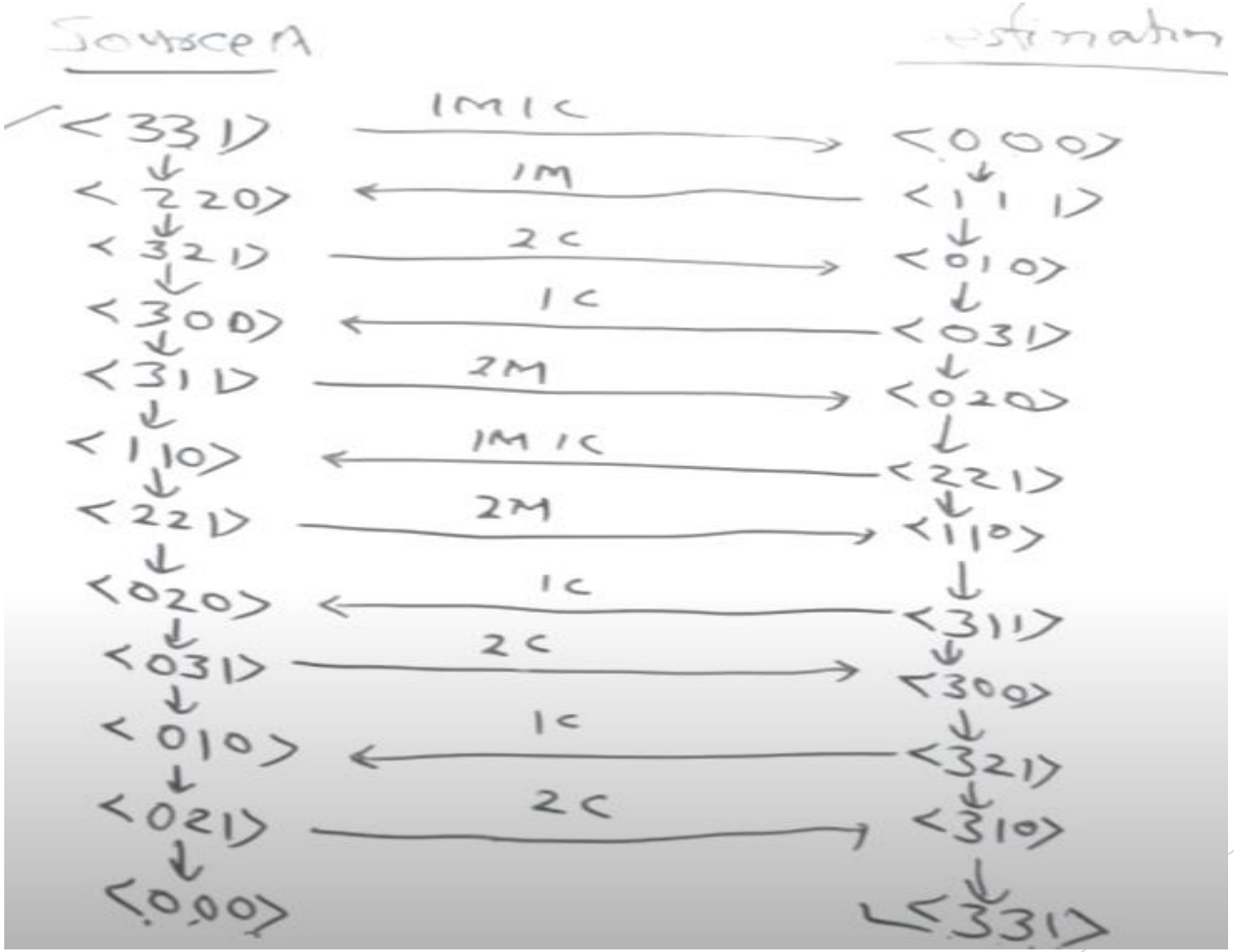
Example of Missionaries and Cannibals Problem

- ˜ The problem statement as discussed in the previous section. Let's formulate the problem first.
- ˜ **States :** In this problem, state can be data structure having triplet (i,j,k) representing the number of missionaries, cannibals, and canoes on the left bank of the river respectively.
- **1. Initial state :** It is (3, 3, 1), as all missionaries, cannibals and canoes are on the left bank of the river.
- **2. Actions :** Take x number of missionaries and y number of cannibals
- **3. Successor function :** If we take one missionary, one cannibal the other side of the river will have two missionaries and two cannibals left.
- **4. Goal test :** Reached state (0, 0, 0)
- **5. Path cost :** Number of crossings to attain the goal state.

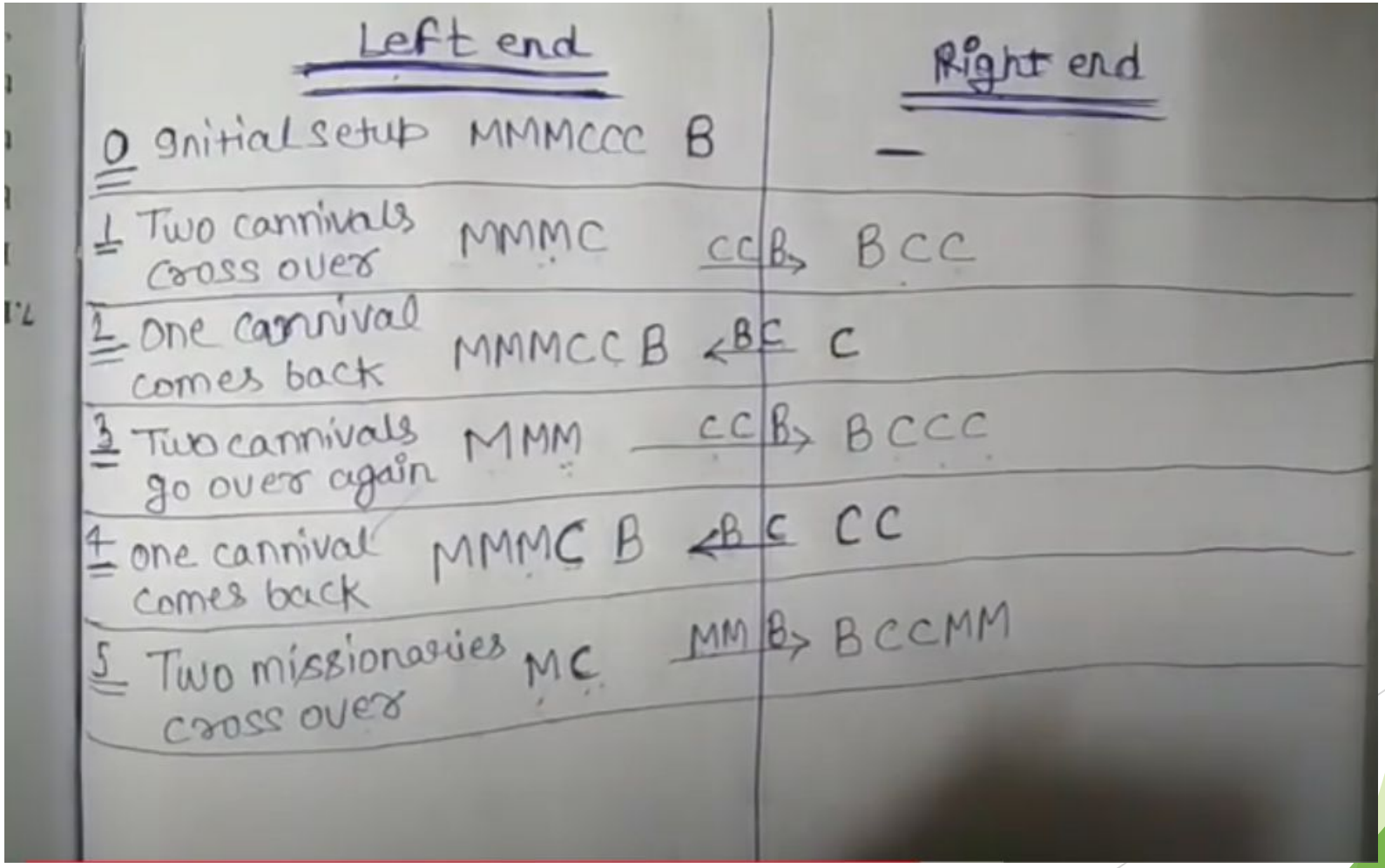
Example of Missionaries and Cannibals Problem



<M,C,B>

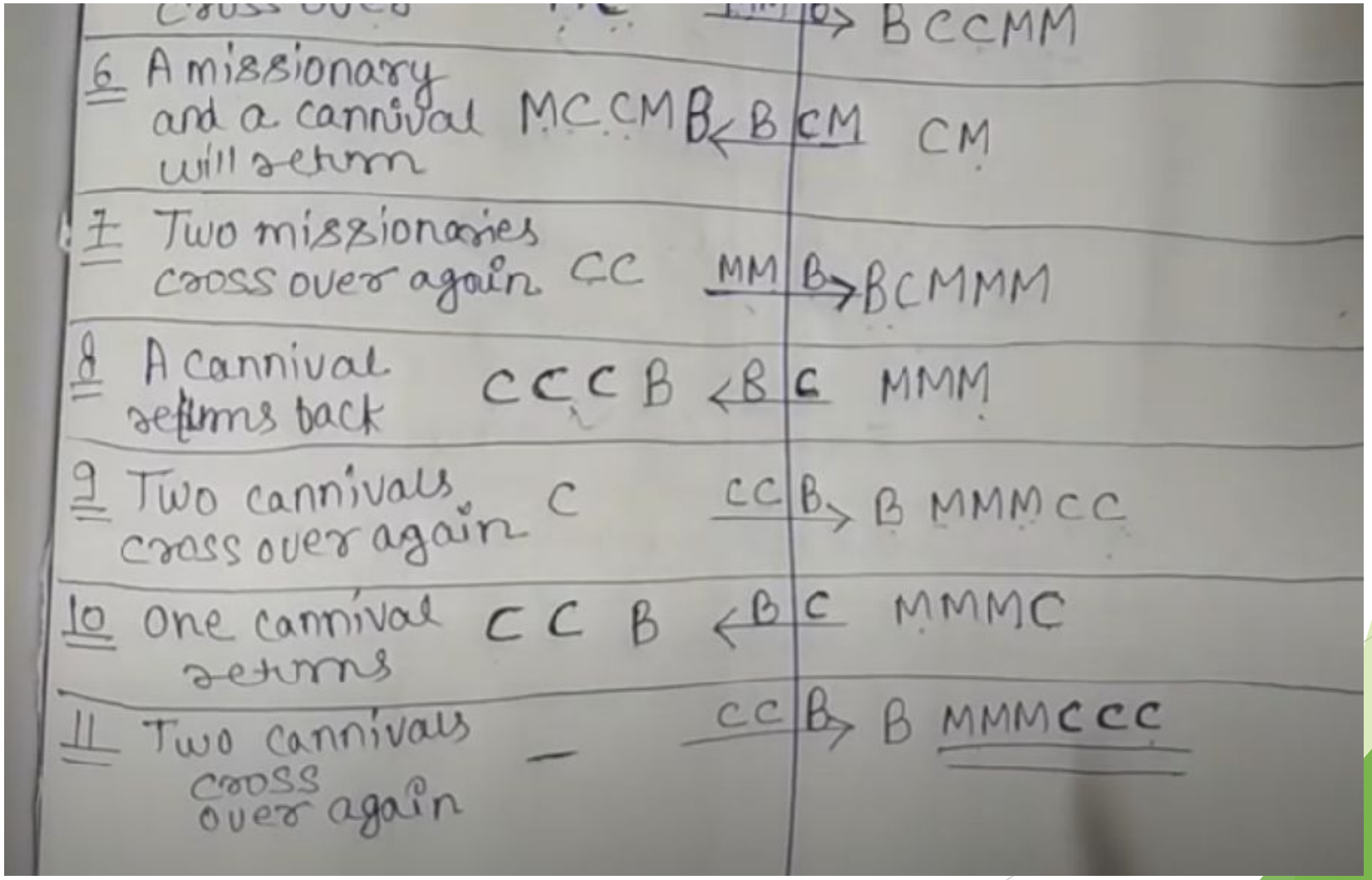

Source A
Destination
<3,3,1>
1M 1C
<0,0,0>
1M
<2,2,0>
<1,1,1>
2C
<3,2,1>
<0,1,0>
1C
<3,0,0>
<0,3,1>
2M
<3,1,1>
<0,2,0>
1M 1C
<1,1,0>
<2,2,1>
2M
<2,2,1>
<1,1,0>
1C
<0,2,0>
<3,1,1>
2C
<0,3,1>
<3,0,0>
1C
<0,1,0>
<3,2,1>
2C
<0,2,1>
<3,1,0>
<0,0,0>
<3,3,1>

Example of Missionaries and Cannibals Problem



	Left end		Right end
0	Initial setup	MMMCCC B	—
1	Two cannivals cross over	MMMC	CC B → B CC
2	One cannival comes back	MMMCC B	← B C C
3	Two cannivals go over again	MMM	CC B → B CCC
4	one cannival comes back	MMMC B	← B C CC
5	Two missionaries cross over	MC	MM B → B CCMM

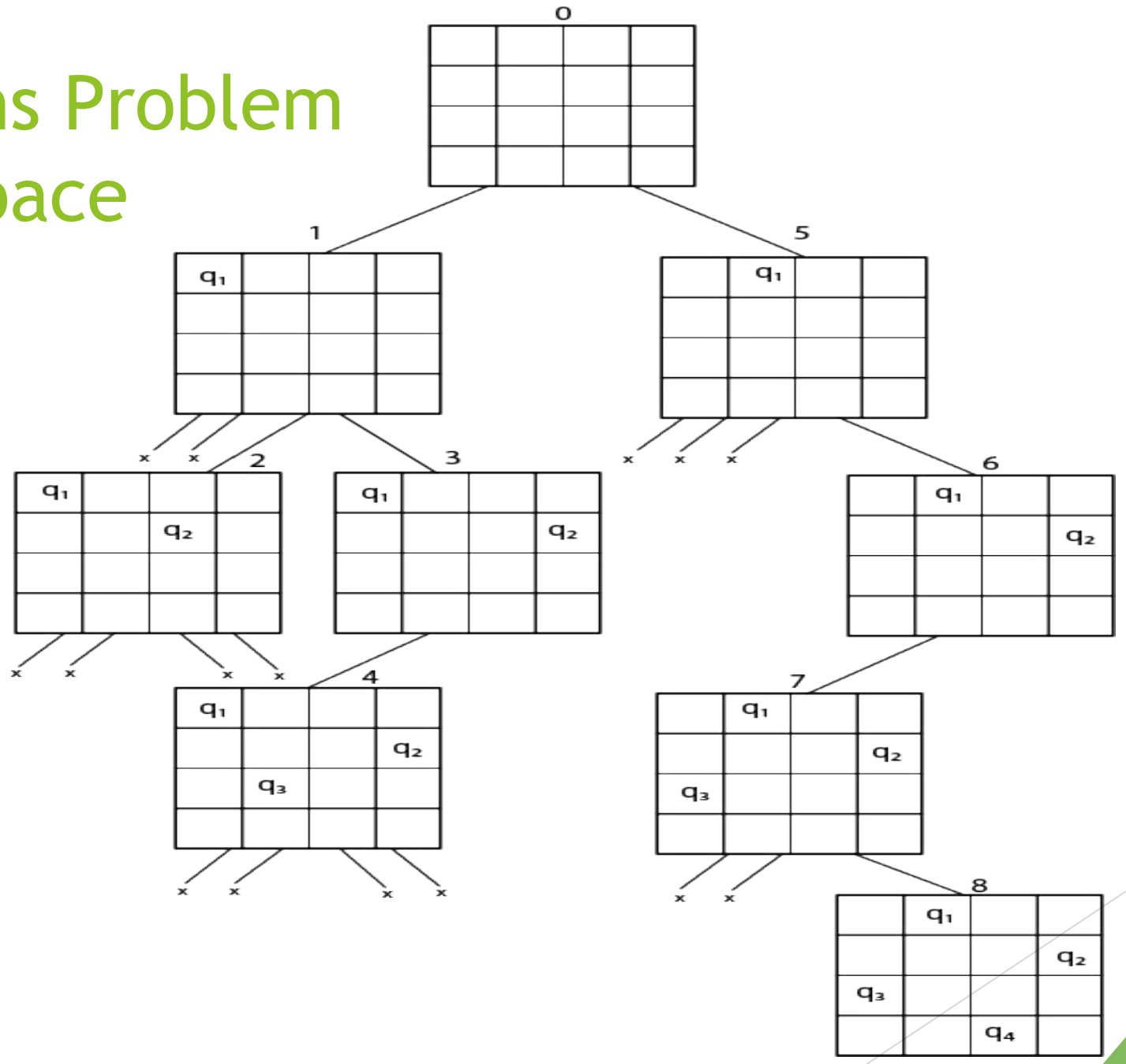
Example of Missionaries and Cannibals Problem

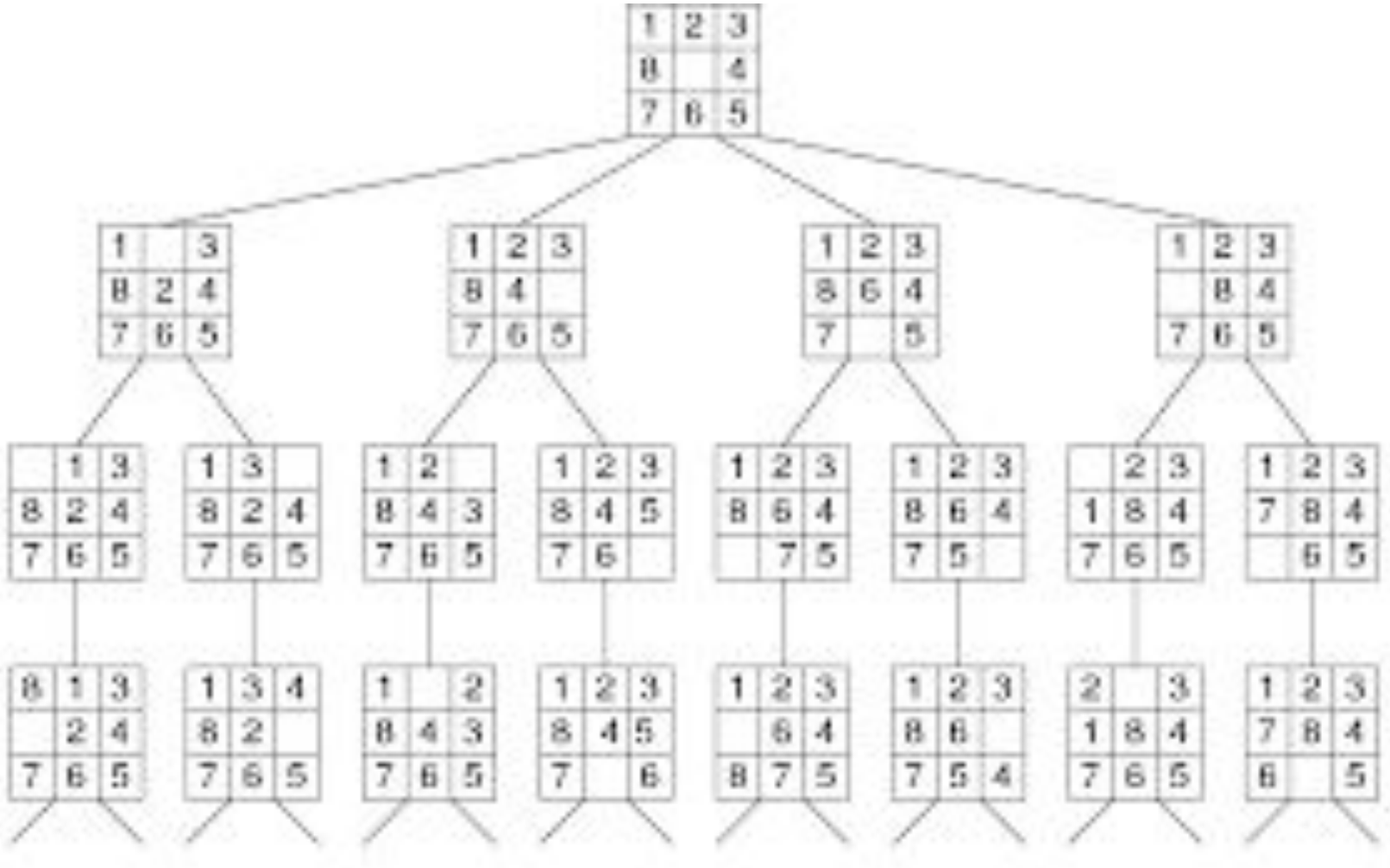


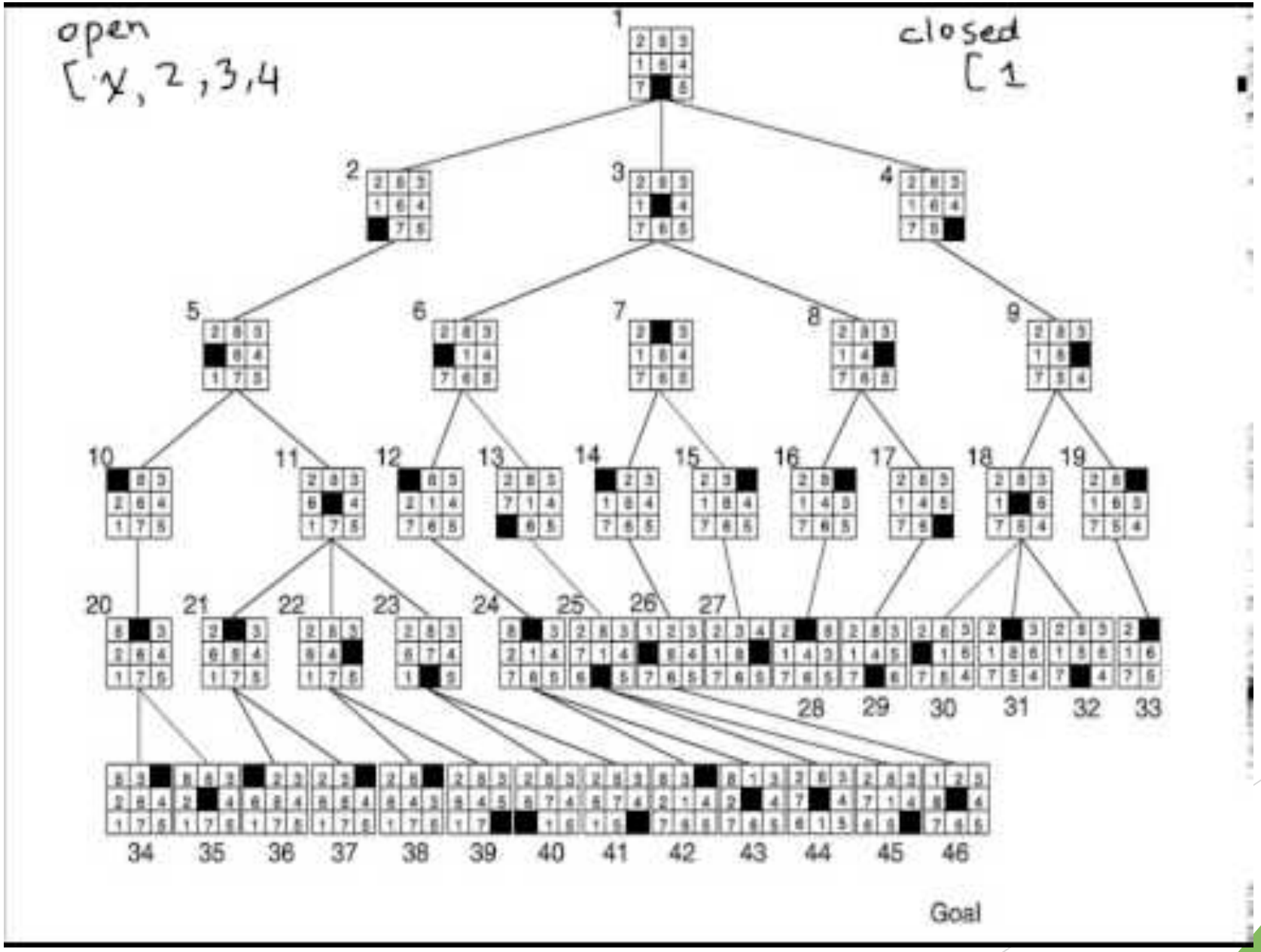
Example of Missionaries and Cannibals Problem

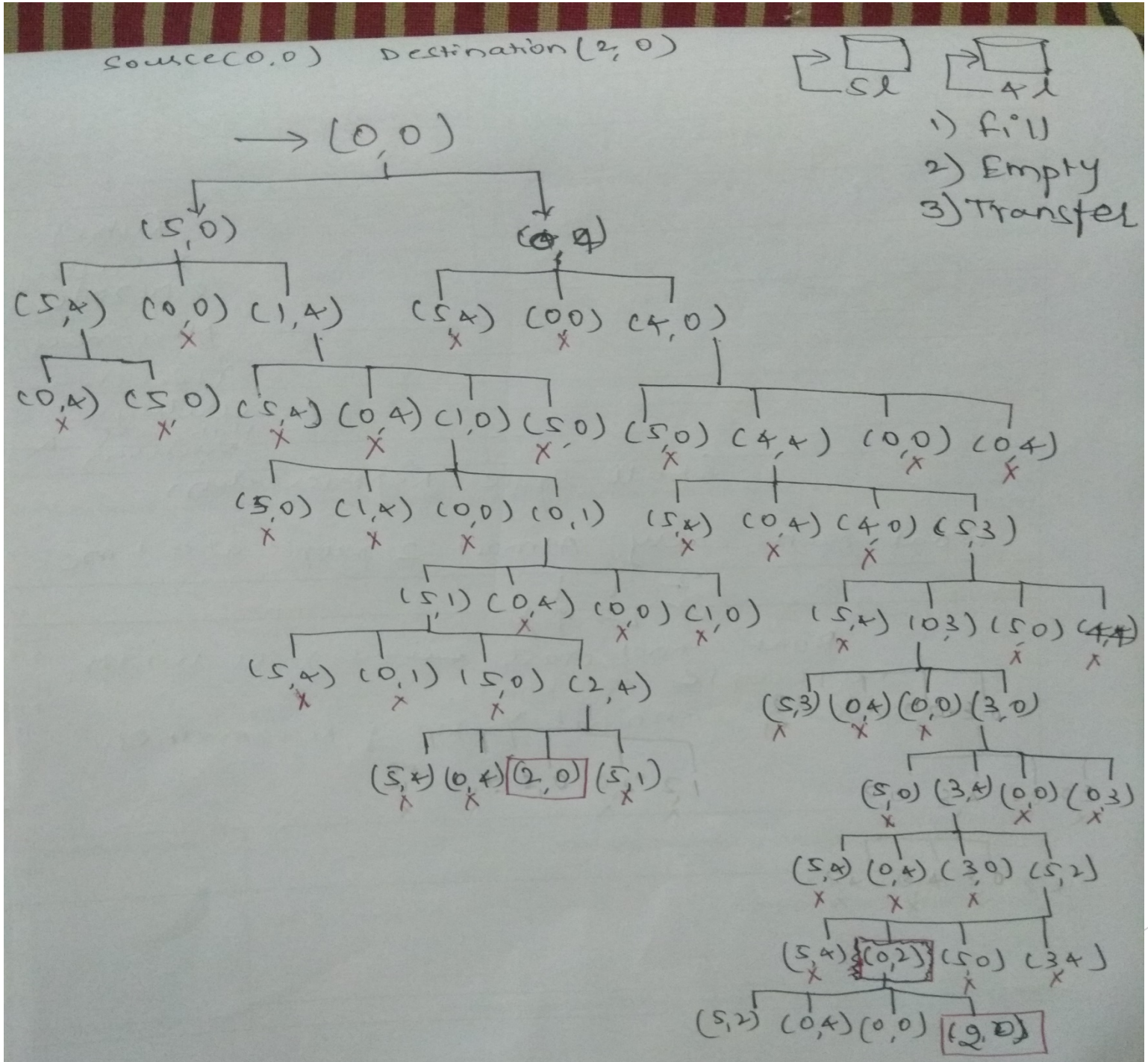
- **Solution :**
- The sequence of actions within the path :
- (3,3,1) → (2,2,0) →(3,2,1) → (3,0,0) → (3,1,1) → (1,1,0) → (2,2,1) → (0,2,0) → (0,3,1) → (0,1,0) → (0,2,1) → (0,0,0)
- Cost = 11 crossings

4 Queens Problem State Space






open
[x, 2, 3, 4
closed
[1
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
Goal


Source(0,0) Destination(2,0)
5l
4l
1) Fill
2) Empty
3) Transfer
(0,0)
(5,0) (0,4)
(5,4) (0,0) (1,4) (5,4) (0,0) (4,0)
(0,4) (5,0) (5,4) (0,4) (1,0) (5,0) (5,0) (4,4) (0,0) (0,4)
(5,0) (1,4) (0,0) (0,1) (5,4) (0,4) (4,0) (5,3)
(5,1) (0,4) (0,0) (1,0) (5,4) (0,3) (5,0) (4,4)
(5,4) (0,1) (5,0) (2,4) (5,3) (0,4) (0,0) (3,0)
(5,4) (0,4) (2,0) (5,1) (5,0) (3,4) (0,0) (0,3)
(5,4) (0,4) (3,0) (5,2)
(5,4) (0,2) (5,0) (3,4)
(5,2) (0,4) (0,0) (2,0)

Important Questions(Qbank)

- Q. Define artificial intelligence.
- Q. Write a short note on : Applications of artificial intelligence.
- Q. What are the four approaches to define AI
- Q. What is Turing test?
- Q. What are the components of AI?
- Q. Explain PEAS representation with example.
- Q. Define agent and give classification of agents.
- Q. What is intelligent agent?
- Q. Write a short note on: Rational agent.
- Q. Write a short note on : Structure of Intelligent agents.
- Q. Give types of agents.
- Q. What are various agent environments? Give PEAS representation for an agent.
- Q. What do you mean by PEAS? Explain properties of task environment.
- Q. Explain detail architecture of goal based agent.
- Q. Explain detail architecture of Model based agent
- Q. Explain Simple reflex agent architecture.
- Q. Explain learning agent architecture and its components
- Q. Write state space representation for classical AI problems