# UNIT 1

# INT 426- GENERATIVE ARTIFICIAL INTELLIGENCE

Unit I

- Introduction : What Is Intelligence?
- History of AI
- Branches of AI
- Types of AI
- Applications of AI
- Defining Intelligence Using Turing Test
- Making Machine Think Like Human
- Building Rational Agent
- Building an Intelligent Agent
- General Problem Solver
- Problem Spaces and Search
- Problem Characteristics
- Production System and Its Characteristics

# Artificial Intelligence?

**“Our intelligence is what makes us human, and AI is an extension of that quality.”**
**– Yann LeCun**

- **Artificial Intelligence** (**AI**) is a way to make machines think and behave intelligently.

  These machines are controlled by software inside them. So AI has a lot to do with intelligent software programs that control these machines.

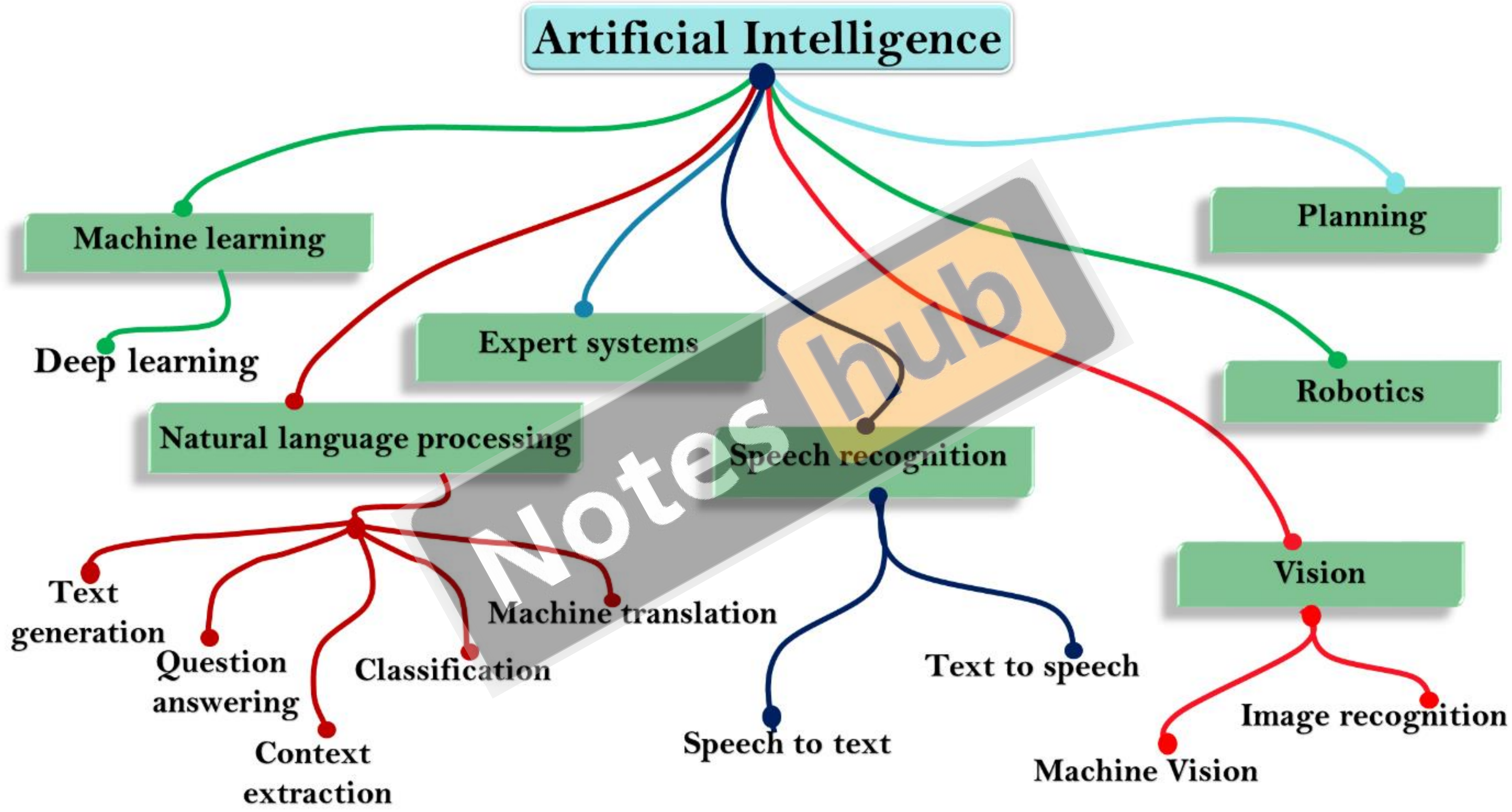
- The design and study of computer programs that react flexibly and intelligently to a wide range of situations [Dean 1995].

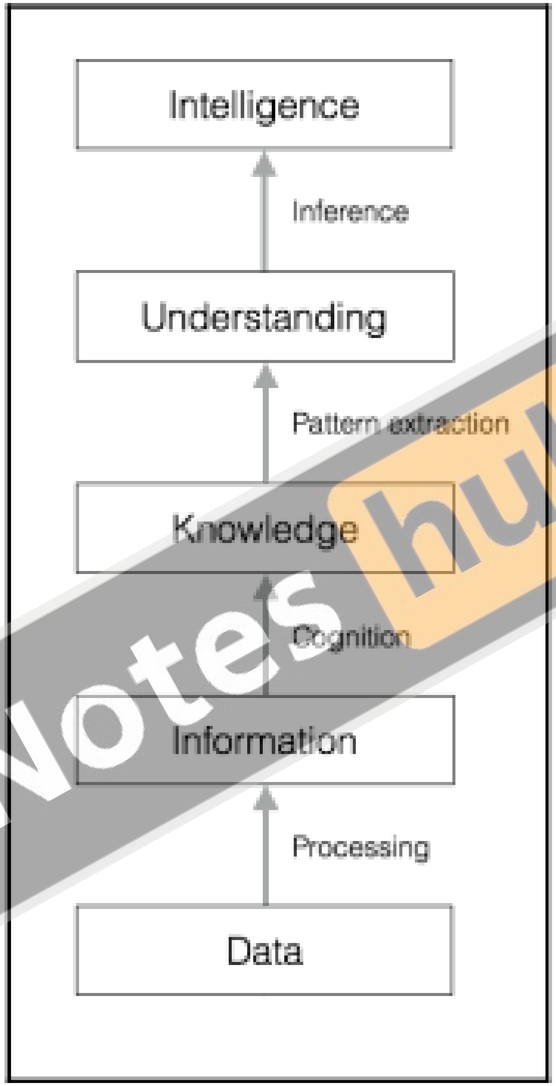
*Dean, Thomas and James Allen and Yiannis Aloimonos. Artificial Intelligence: Theory and Practice. Menlo Park: Addison-Wesley, 1995.*

- "The exciting new effort to make computers think . . . *machines with minds,* in the full and literal sense.. (Haugeland, 1985)
- "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..."(Bellman, 1978)
- "The art of creating machines that perform functions that require intelligence when performed by people" (Kurzweil, 1990)
- "The study of how to make computers do things at which, at the moment, people are better" (Rich and Knight, 199 1 )
- "The study of mental faculties through the use of computational models" (Charniak and McDermott, 1985)
- "The study of the computations that make it possible to perceive, reason, and act" (Winston, 1992)
- "A field of study that seeks to explain and emulate intelligent behavior in terms of computational processes" (Schalkoff, 1 990)
- "The branch of computer science that is concerned with the automation of intelligent behavior" (Luger and Stubblefield, 1993)

**Some definitions of AI. They are organized into four categories**:

| | |
|---|---|
| Systems that think like humans. | Systems that think rationally. |
| Systems that act like humans. | Systems that act rationally. |


Artificial Intelligence
Machine learning
Deep learning
Expert systems
Natural language processing
Text generation
Question answering
Context extraction
Classification
Machine translation
Speech recognition
Speech to text
Text to speech
Planning
Robotics
Vision
Image recognition
Machine Vision



Raw data gets converted to wisdom through various levels of processing

# Why need AI?

One of the main reasons we want to study AI is to automate many things.

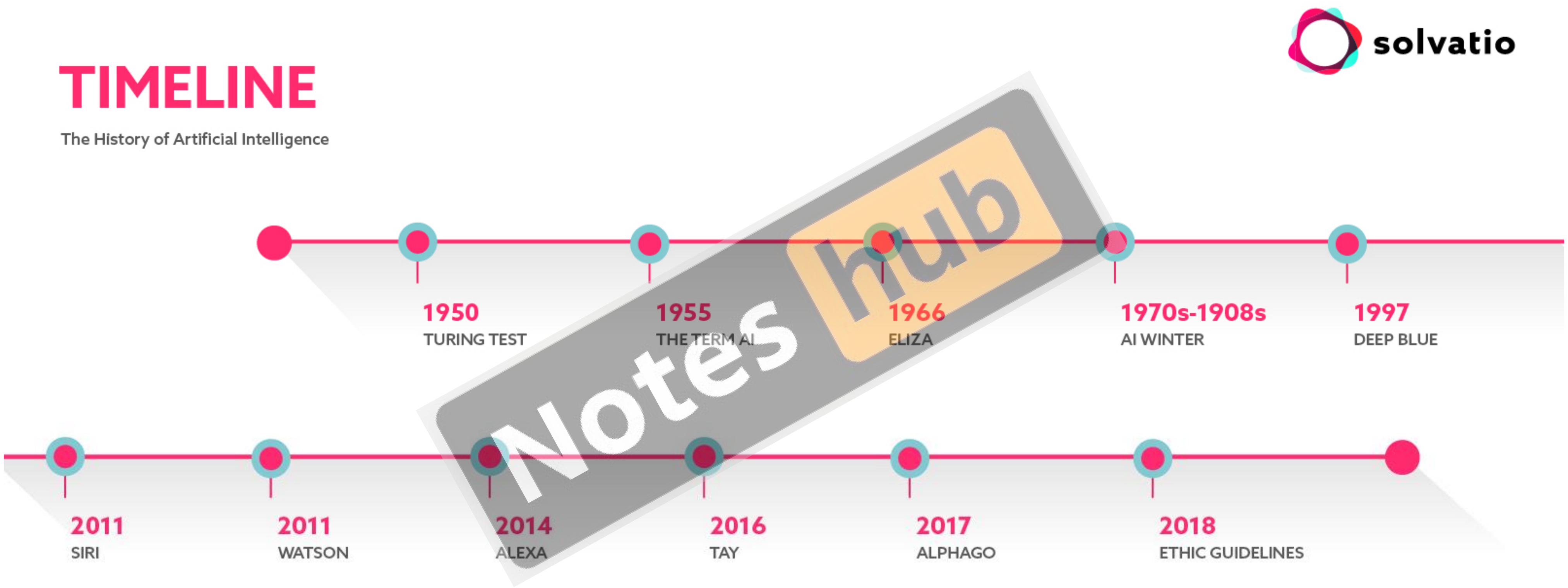
- We deal with huge and insurmountable amounts of data. The human brain can't keep track of so much data.
- Data originates from multiple sources simultaneously.
- The data is unorganized and disordered.
- Knowledge derived from this data has to be updated constantly because the data itself keeps changing.
- The sensing and actuation has to happen in real time with high precision.

Hence, we need to design and develop intelligent machines that can do this. We need AI systems that can:

- Handle large amounts of data in an efficient way. With the advent of Cloud Computing, we are now able to store huge amounts of data.
- Ingest data simultaneously from multiple sources without any lag.
- Index and organize data in a way that allows us to derive insights.
- Learn from new data and update constantly using the right learning algorithms.
- Think and respond to situations based on the conditions in real time.

AI techniques are actively being used to make existing machines smarter, so that they can execute faster and more efficiently.

# The history of artificial intelligence



solvatio

TIMELINE

The History of Artificial Intelligence

1950
TURING TEST

1955
THE TERM AI

1966
ELIZA

1970s-1908s
AI WINTER

1997
DEEP BLUE

2011
SIRI

2011
WATSON

2014
ALEXA

2016
TAY

2017
ALPHAGO

2018
ETHIC GUIDELINES

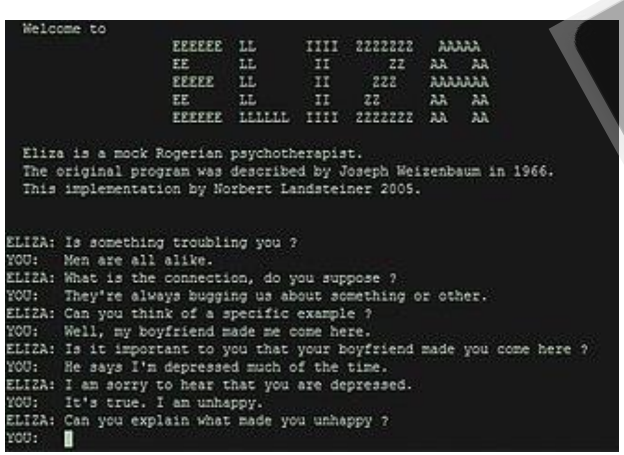
## 1950: Turing Test

Developed by the mathematician Alan Turing, the test finds out whether a machine's ability to think is equivalent to, or indistinguishable from, that of a human

## 1955: The term 'Artificial Intelligence'

The term ‘Artificial Intelligence’ was coined in 1955. The US-American computer scientist John McCarthy first used this term in an application for funding for the first conference on artificial intelligence at Dartmouth College.

## 1966: Eliza the chatbot



```
  Welcome to
                    EEEEEE  LL      IIII  ZZZZZZZ   AAAAA
                    EE      LL       II       ZZ   AA   AA
                    EEEEE   LL       II     ZZZ    AAAAAAA
                    EE      LL       II    ZZ      AA   AA
                    EEEEEE  LLLLLL  IIII  ZZZZZZZ  AA   AA

  Eliza is a mock Rogerian psychotherapist.
  The original program was described by Joseph Weizenbaum in 1966.
  This implementation by Norbert Landsteiner 2005.


ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:   █
```

***Source: https://en.wikipedia.org/wiki/ELIZA***

Eliza is presumably one of the first chatbots.

## 1970s to 1980s: AI winter

After the boom of the early research projects, no new software's and AI programs were developed in the 70s and 80s. *The term AI winter refers to a period of reduced funding in the development of AI*

## 1997: Deep Blue

Deep Blue, the IBM chess computer, won a tournament of six games against the then world chess champion Garri Kasparov – one year before, the computer had still lost against him

***Source: https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer)***

**2011: Siri**

Apple gave its iPhone 4s an intelligent language-based assistant: Siri. The software recognizes and processes natural language and can thus act as a personal assistant.  Siri provides answers and executes commands which the user speaks instead of entering them via the keyboard.

**2011: Watson**

The so-called Supercomputer Watson, named after IBM founder Thomas J. Watson. became famous all over the world when it won the US-American quiz show Jeopardy against two human competitors.

**2014: Alexa**

With Amazon's Alexa, another intelligent language-based assistant is invited into the kitchens, living rooms and bedrooms of many people.

**2016: Tay (bot)**

**Tay** was an artificial intelligence chatter bot that was originally released by Microsoft Corporation via Twitter.

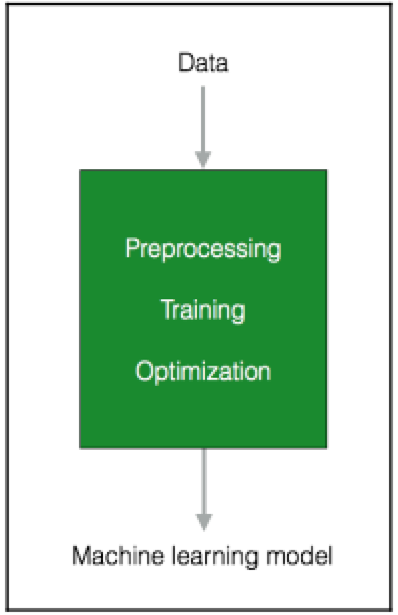
**2017: AlphaGO**

**Developed by Google**, the program AlphaGO beat the world's best go player Ke Jie after having won against Lee Sedol in the previous year. Since the game of go is much more complex than chess, it had been considered impossible to master for a computer.

# Branches of AI

It is important to understand the various fields of study within AI so that we can choose the right framework to solve a given real-world problem.

- **Machine learning and pattern recognition**:
  - We design and develop software that can learn from data. Based on these learning models, we perform predictions on unknown data.
  - One of the main constraints here is that these programs are limited to the power of the data. If the dataset is small, then the learning models would be limited as well.
  - For example, in a face recognition system, the software will try to match the pattern of eyes, nose, lips, eyebrows, and so on in order to find a face in the existing database of users.



- **Logic-based AI**:

  Mathematical logic is used to execute computer programs in logic-based AI. A program written in logic-based AI is basically a set of statements in logical form that express facts and rules about a particular problem domain.
  This is used extensively in pattern matching, language parsing, semantic analysis, and so on.

- **Search**:

  The Search techniques are used extensively in AI programs. These programs examine a large number of possibilities and then pick the most optimal path.
  For example, this is used a lot in strategy games such as Chess, networking, resource allocation, scheduling, and so on.

- **Knowledge representation**:

  The facts about the world around us need to be represented in some way for a system to make sense of them.
  If knowledge is represented efficiently, systems can be smarter and more intelligent.

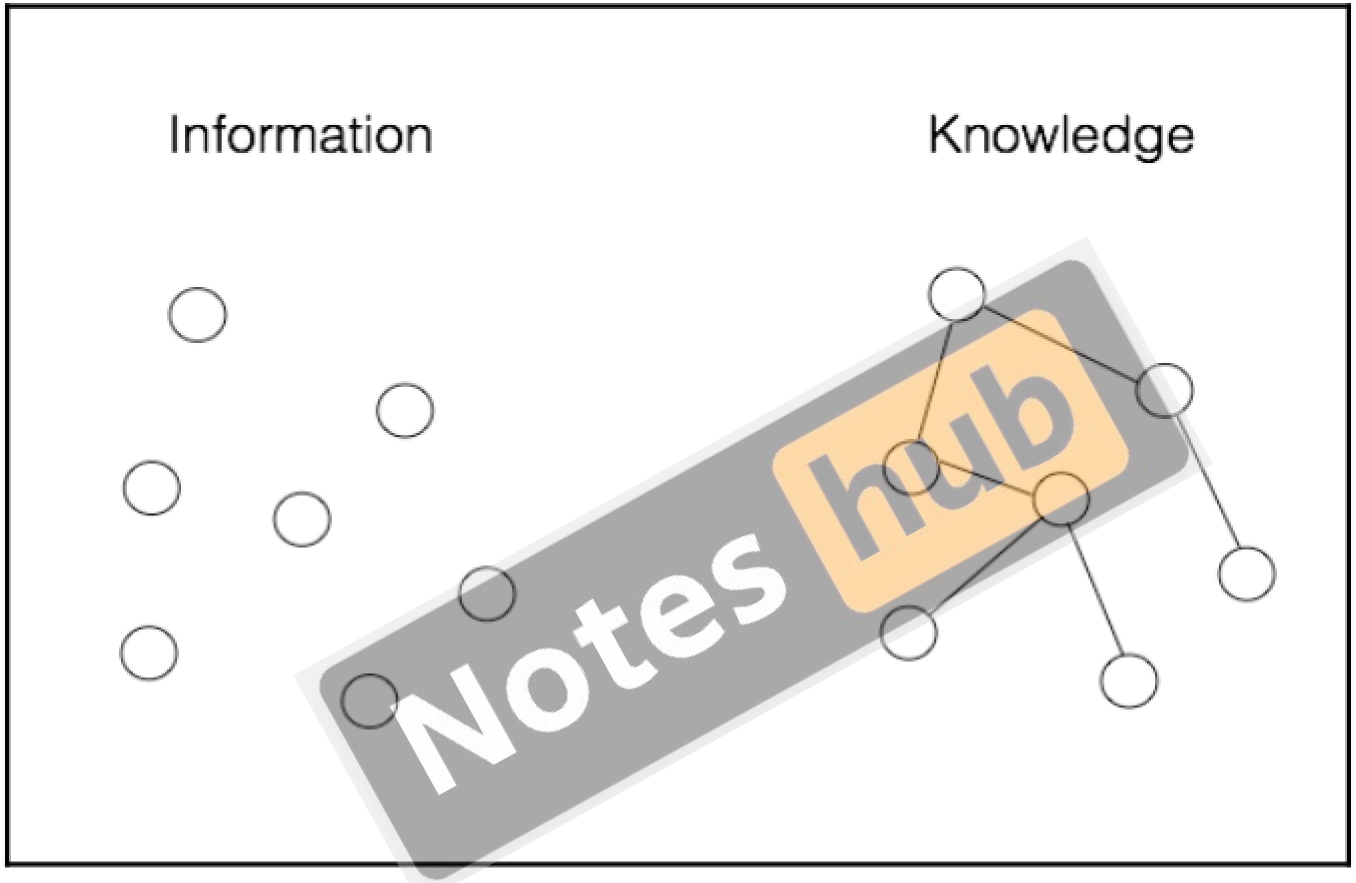
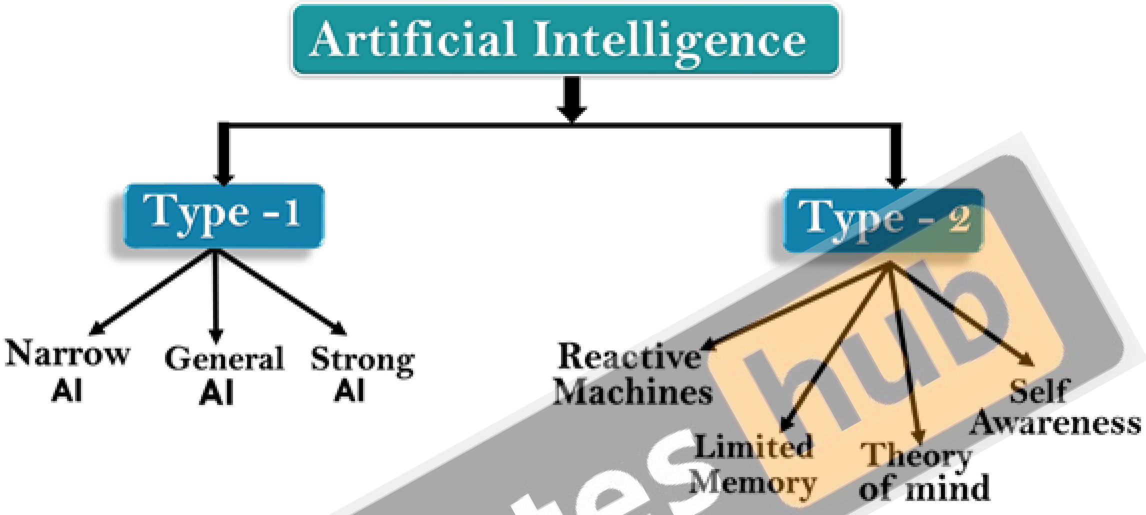


Diagram shows the difference between information and knowledge:

# Types of Artificial Intelligence:



## Weak AI or Narrow AI:

- Narrow AI is a type of AI which is able to perform a dedicated task with intelligence. The most common and currently available AI is Narrow AI in the world of Artificial Intelligence.
- Narrow AI cannot perform beyond its field or limitations, as it is only trained for one specific task. Hence it is also termed as weak AI.
- Some Examples of Narrow AI are playing chess, purchasing suggestions on e-commerce site, speech recognition, and image recognition.

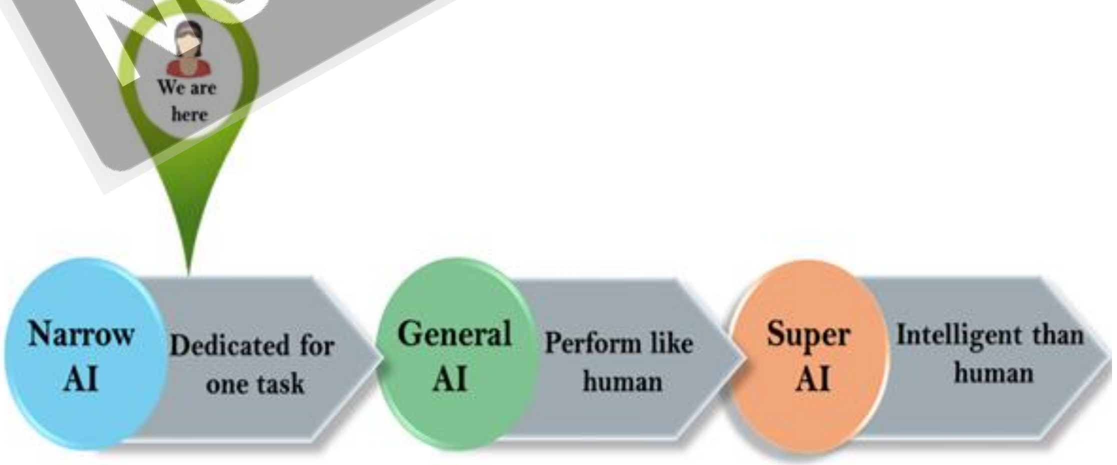
## General AI:

- General AI is a type of intelligence which could perform any intellectual task with efficiency like a human.
- The idea behind the general AI to make such a system which could be smarter and think like a human by its own.

systems with general AI are still under research

## Super AI:

Super AI is a level of Intelligence of Systems at which machines could surpass human intelligence, and can perform any task better than human with cognitive properties. It is an outcome of general AI



# Artificial Intelligence type-2

## Reactive Machines

- Purely reactive machines are the most basic types of Artificial Intelligence.
- Such AI systems do not store memories or past experiences for future actions.
- These machines only focus on current scenarios and react on it as per possible best action.
- IBM's Deep Blue system and Google's AlphaGo is also an example of reactive machines.

## Limited Memory

- Limited memory machines can store past experiences or some data for a short period of time.
- These machines can use stored data for a limited time period only.
- Self-driving cars are one of the best examples of Limited Memory systems. These cars can store recent speed of nearby cars, the distance of other cars, speed limit, and other information to navigate the road.

## Theory of Mind

- Theory of Mind AI should understand the human emotions, people, beliefs, and be able to interact socially like humans.

This type of AI machines are still not developed

## Self-Awareness

- Self-awareness AI is the future of Artificial Intelligence. These machines will be super intelligent, and will have their own consciousness, sentiments, and self-awareness.
- These machines will be smarter than human mind.
- Self-Awareness AI is a hypothetical concept.

# Intelligent Systems in Your Everyday Life

- Post Office
  - automatic address recognition and sorting of mail
- Banks
  - automatic check readers, signature verification systems
  - automated loan application classification
- Telephone Companies
  - automatic voice recognition for directory inquiries
  - automatic fraud detection,
  - classification of phone numbers into groups
- Credit Card Companies
  - automated fraud detection, automated screening of applications
- Computer Companies
  - automated diagnosis for help-desk applications

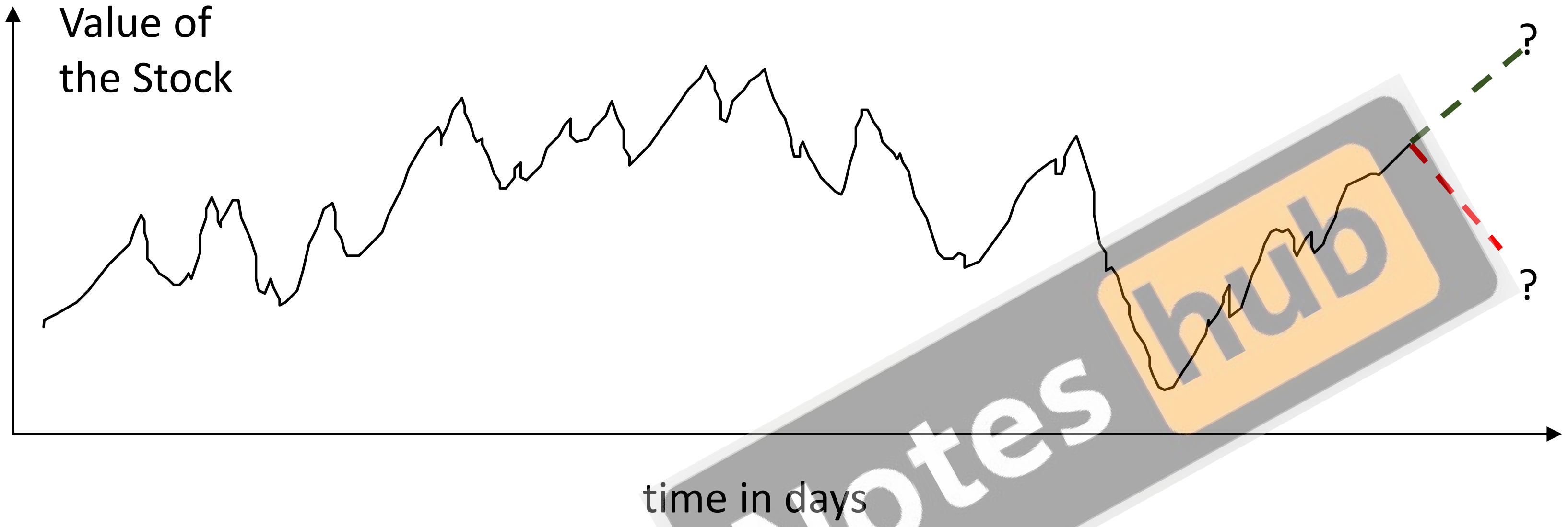
# AI Applications: Consumer Marketing

- Have you ever used any kind of credit/ATM/store card while shopping?
  - if so, you have very likely been “input” to an AI algorithm
- All of this information is recorded digitally
- Companies like **Nielsen** gather this information weekly and search for patterns
  - general changes in consumer behavior
  - tracking responses to new products
  - identifying customer segments: targeted marketing, e.g., they find out that consumers with sports cars who buy textbooks respond well to offers of new credit cards.
  - Currently a very hot area in marketing
- How do they do this?
  - Algorithms (“data mining”) search data for patterns
  - based on mathematical theories of learning
  - completely impractical to do manually

# AI Applications: Identification Technologies

- ID cards
  - e.g., ATM cards
  - can be a trouble and security risk:
    - cards can be lost, stolen, passwords forgotten, etc
- Biometric Identification
  - walk up to a locked door
    - camera
    - fingerprint device
    - microphone
  - computer uses your biometric signature for identification
    - face, eyes, fingerprints, voice pattern

# AI Applications: Predicting the Stock Market



- The Prediction Problem
  - given the past, predict the future
  - very difficult problem!
  - we can use learning algorithms to learn a predictive model from historical data
    - `prob(increase at day t+1 | values at day t, t-1,t-2....,t-k)`
  - such models are routinely used by banks and financial traders to manage portfolios worth millions of dollars

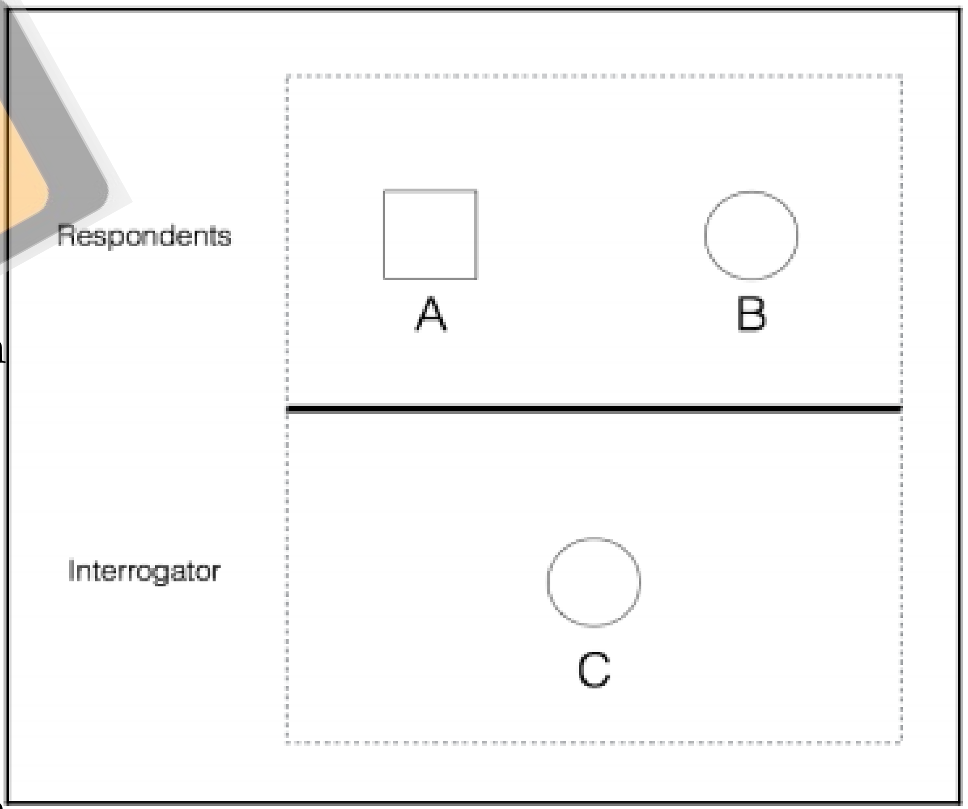
# AI-Applications: Machine Translation

- Language problems in international business
  - e.g., at a meeting of Japanese, Korean, Vietnamese and Swedish investors, no common language
  - or: you are shipping your software manuals to 127 countries
  - solution; hire translators to translate
  - would be much cheaper if a machine could do this!

- How hard is automated translation
  - very difficult!
  - e.g., English to Russian
    - “The spirit is willing but the flesh is weak” (English)
    - “the vodka is good but the meat is rotten” (Russian)
  - not only must the words be translated, but their meaning also!

# Defining intelligence using Turing Test
# ( How Intelligence is Defined)

- The legendary computer scientist and mathematician, ***Alan Turing***, proposed the Turing Test to define a definition of intelligence.
- it is a test to see if a computer can learn to mimic human behavior.
- He defined intelligent behavior as the ability to achieve human-level intelligence during a conversation.



## Test Setup

- He proposed that a human should interrogate the machine through a text interface.
- Another constraint is that the human cannot know who's on the other side of the interrogation, which means it can either be a machine or a human.
- To enable this setup, a human will be interacting with two entities through a text interface.
- These two entities are called respondents.
- One of them will be a human and the other one will be the machine.
- **The respondent machine passes the test if the interrogator is unable to tell whether the answers are coming from a machine or a human.**

It’s a quite a difficult task for the respondent machine because lot of things going on during a conversation.

## Thinks with which machine need to be competent

1. **Natural Language Processing.: The** machine needs this to communicate with the interrogator. The machine needs to parse the sentence, extract the context, and give an appropriate answer.

2. **Knowledge Representation**: The machine needs to store the information provided before the interrogation. It also needs to keep track of the information being provided during the conversation so that it can respond appropriately if it comes up again.

3. **Reasoning**: It's important for the machine to understand how to interpret the information that gets stored.
   Humans tend to do this automatically to draw conclusions in real time.

4. **Machine Learning**: This is needed so that the machine can adapt to new conditions in real time. The machine needs to analyze and detect patterns so that it can draw inferences.

There is another thing called the Total Turing Test that deals with vision and movement.
To pass this test, the machine needs to see objects using computer vision and move around using Robotics.

# Making machines think like humans

For decades, we have been trying to get the machine to think like a human.
In order to make this happen, we need to understand how humans think in the first place.
***How do we understand the nature of human thinking?***

1. One way to do this would be to note down how we respond to things. But this quickly becomes intractable, because there are too many things to note down.
2. Another way to do this is to conduct an experiment based on a predefined format. We develop a certain number of questions to encompass a wide variety of human topics, and then see how people respond to it.

Once we gather enough data, **we can create a model to simulate the human process.**
**This model can be used to create software that can think like humans.**

- ✓ All we care about is the output of the program given a particular input.
- ✓ If the program behaves in a way that matches human behavior, then we can say that humans have a similar thinking mechanism.

- **Cognitive Modeling is the field of computer science which** deals with simulating the human thinking process.
- It tries to understand how humans solve problems.
- it takes the mental processes that go into this problem solving process and turns it into a software model.
- This model can then be used to simulate human behavior.

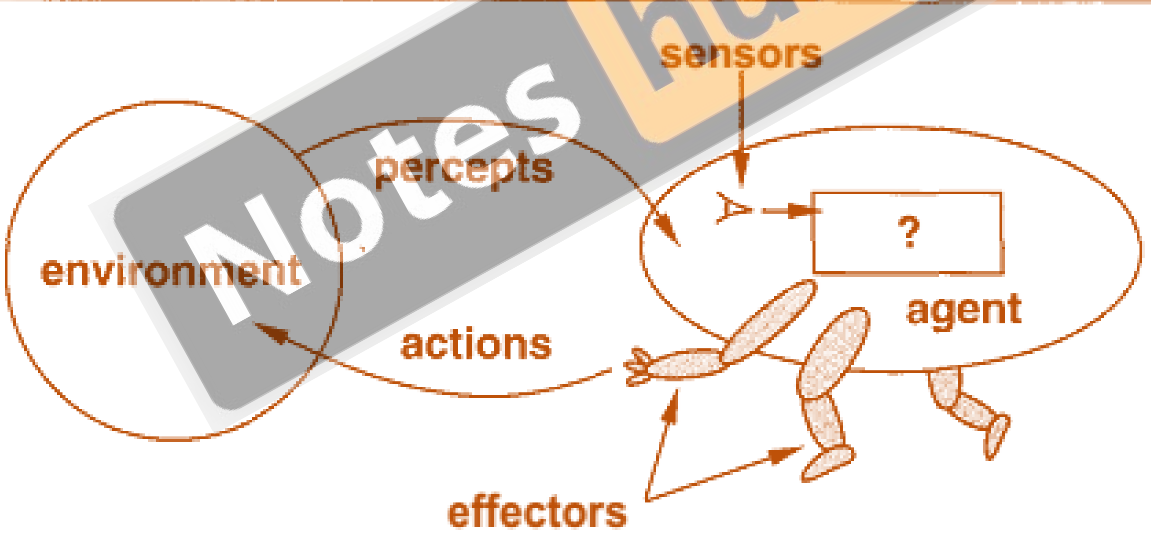
- *Cognitive modeling is used in a variety of AI applications such as deep learning, expert systems, Natural Language Processing, robotics, and so on.*

# INTELLIGENT AGENTS

## Introduction:

- **An agent is** anything that can be viewed as **perceiving** its environment through **sensors and acting** upon that environment through **effectors.**

- A human agent has eyes, ears, and other organs for sensors, and hands, legs, mouth, and other body parts for effectors.
- A robotic agent substitutes cameras and infrared range finders for the sensors and various motors for the effectors.
- Software agent can have keystrokes, file contents as sensory input and act on those inputs and display output on the screen.



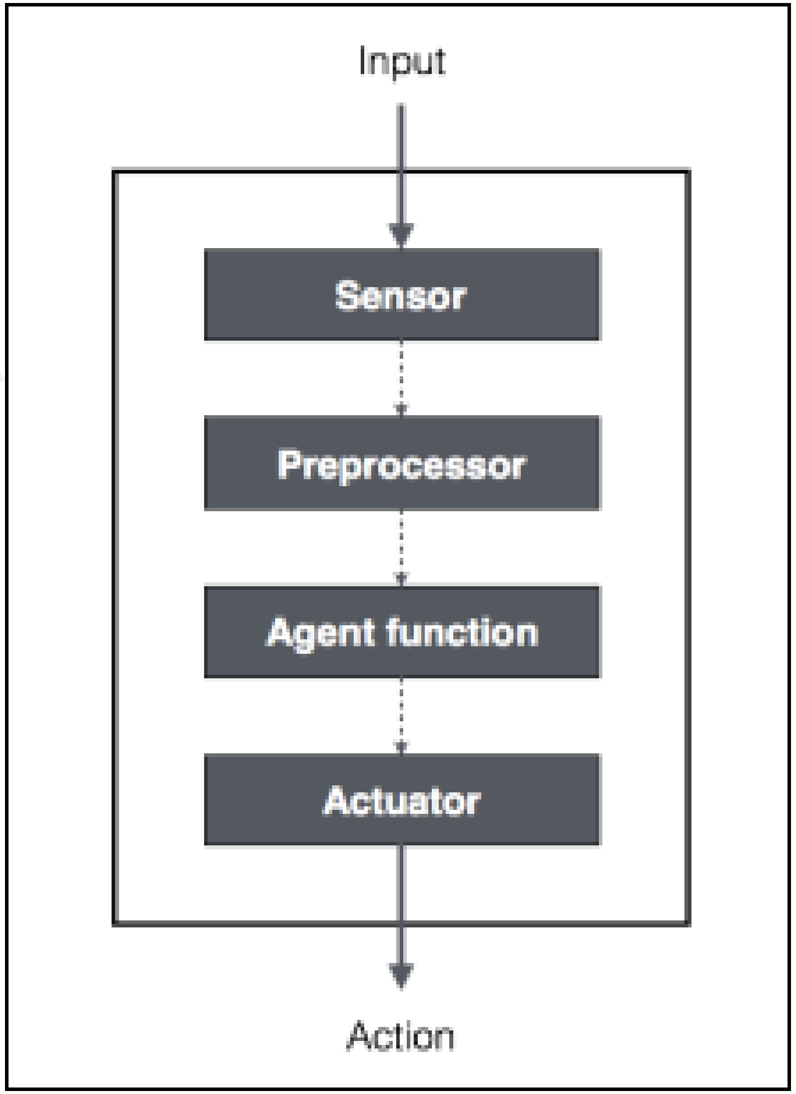
*Agents interact with environments through sensors and effectors.*

***Aim:* 1.** *Design agents that do **a good job** of acting on their environment.*

# Rational agents

***What is Rational?***

- Rationality refers to doing the right thing in a given circumstance.
- This needs to be performed in such a way that there is maximum benefit to the entity performing the action.
- An agent is said to act rationally if,  given a set of rules, it takes actions to achieve its goals.



# STRUCTURE OF INTELLIGENT AGENTS

An agents is described by their *behavior*—the action that is performed after any given sequence of percepts .

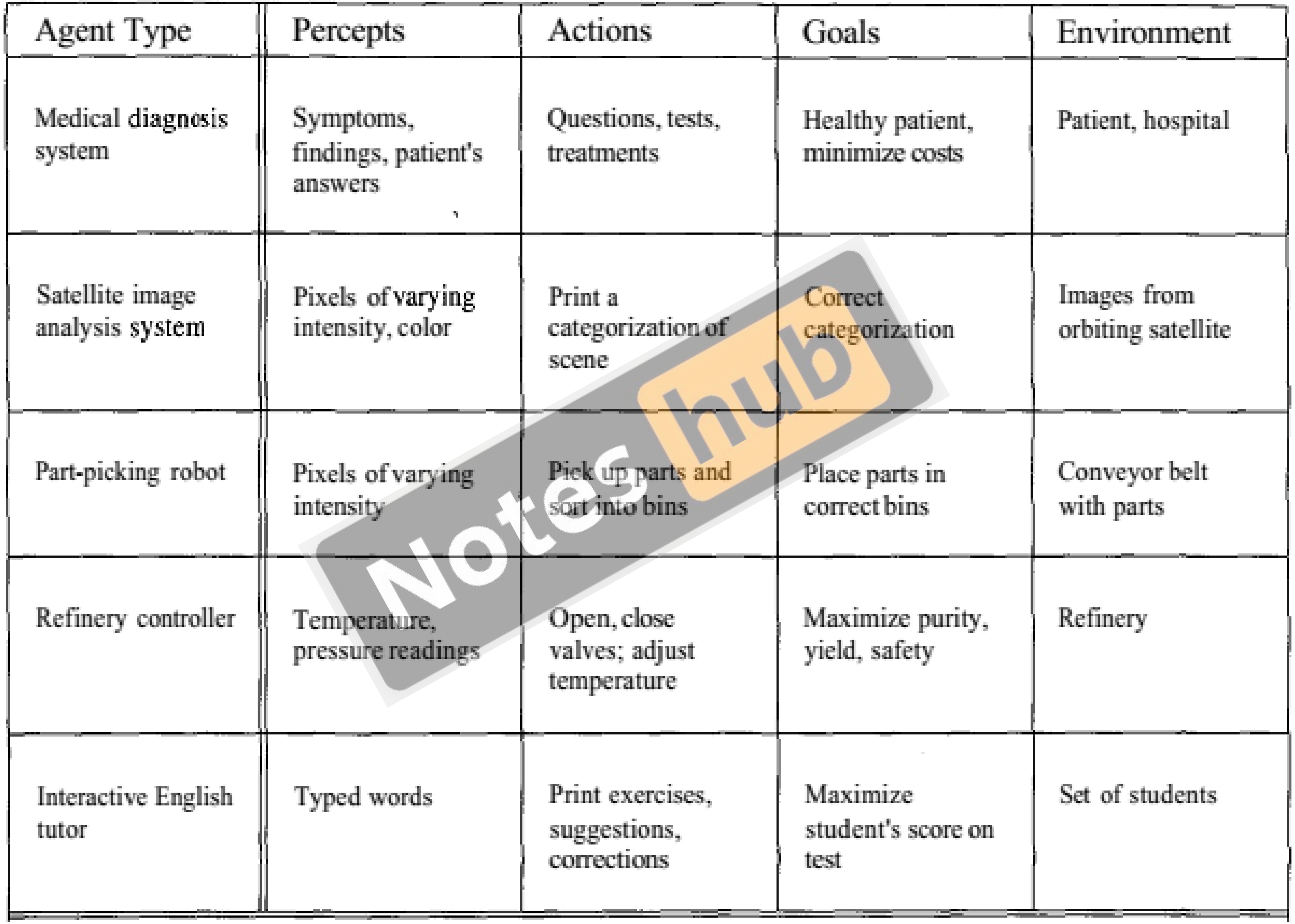
- The job of AI is to design the **agent program:** a function that implements the agent mapping from percepts to actions.

- We assume this program will run on some sort of ARCHITECTURE computing device, which we will call the **architecture.** The architecture might be a plain computer, or it might include special-purpose hardware for certain tasks, such as processing camera images or filtering audio input.

*The relationship among agents, architectures, and programs can be as follows:*

***agent = architecture + program***

***Before we design an agent program, we must have a pretty good idea of the possible percepts and actions, what goals or performance measure the agent is supposed to achieve, and what sort of environment it will operate in (PAGE).*** Percepts, Actions, Goals, Environment (PAGE)



| Agent Type | Percepts | Actions | Goals | Environment |
|---|---|---|---|---|
| Medical diagnosis system | Symptoms, findings, patient's answers | Questions, tests, treatments | Healthy patient, minimize costs | Patient, hospital |
| Satellite image analysis system | Pixels of varying intensity, color | Print a categorization of scene | Correct categorization | Images from orbiting satellite |
| Part-picking robot | Pixels of varying intensity | Pick up parts and sort into bins | Place parts in correct bins | Conveyor belt with parts |
| Refinery controller | Temperature, pressure readings | Open, close valves; adjust temperature | Maximize purity, yield, safety | Refinery |
| Interactive English tutor | Typed words | Print exercises, suggestions, corrections | Maximize student's score on test | Set of students |

## Types / Classes of Agents

1. Simple Reflex Agents
2. Model-Based Reflex Agents
3. Goal-Based Agents
4. Utility-Based Agents

# *1. Simple Reflex Agents*

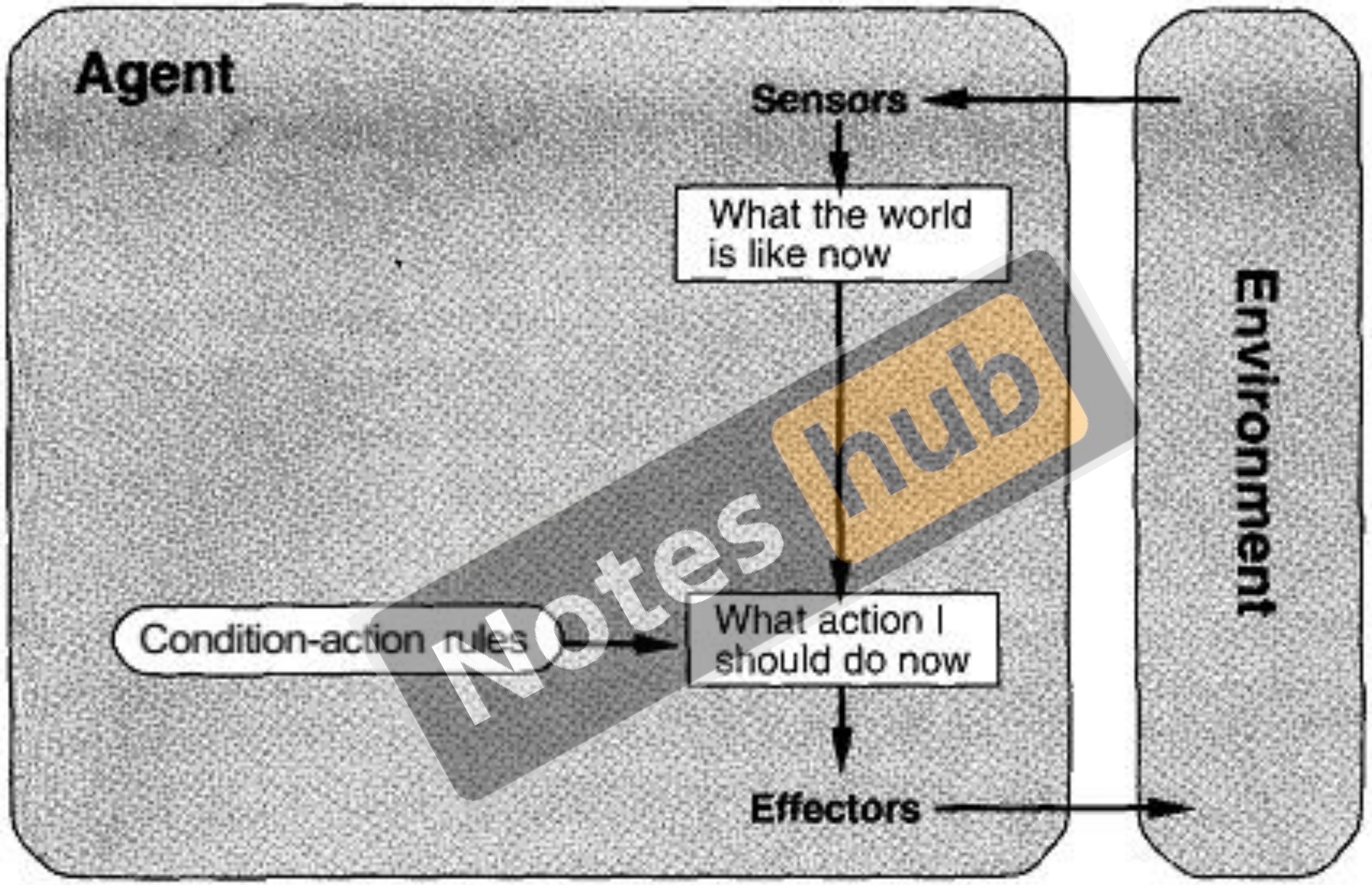
- This agent selects actions based on the agents current perception or the world and not based on past perceptions.
    - For example if a mars lander found a rock in a specific place it needed to collect then it would collect it, if it was a simple reflex agent then if it found the same rock in a different place it would still pick it up as it doesn't take into account that it already picked it up.
- This is *useful for when a quick automated response is needed*, humans have a very similar reaction to fire for example, our brain pulls our hand away without thinking about any possibility that there could be danger in the path of your arm. We call these reflex actions.
- The agent function is based on the condition-action rule also called as Situation-action rules, or if-then rules.

**if** *car-in-front-is-braking***then** *initiate-braking*

**if** hand is in fire **then** pull away hand

✓ A condition-action rule is a rule that maps a state i.e, condition to an action. If the condition is true, then the action is taken, else not.

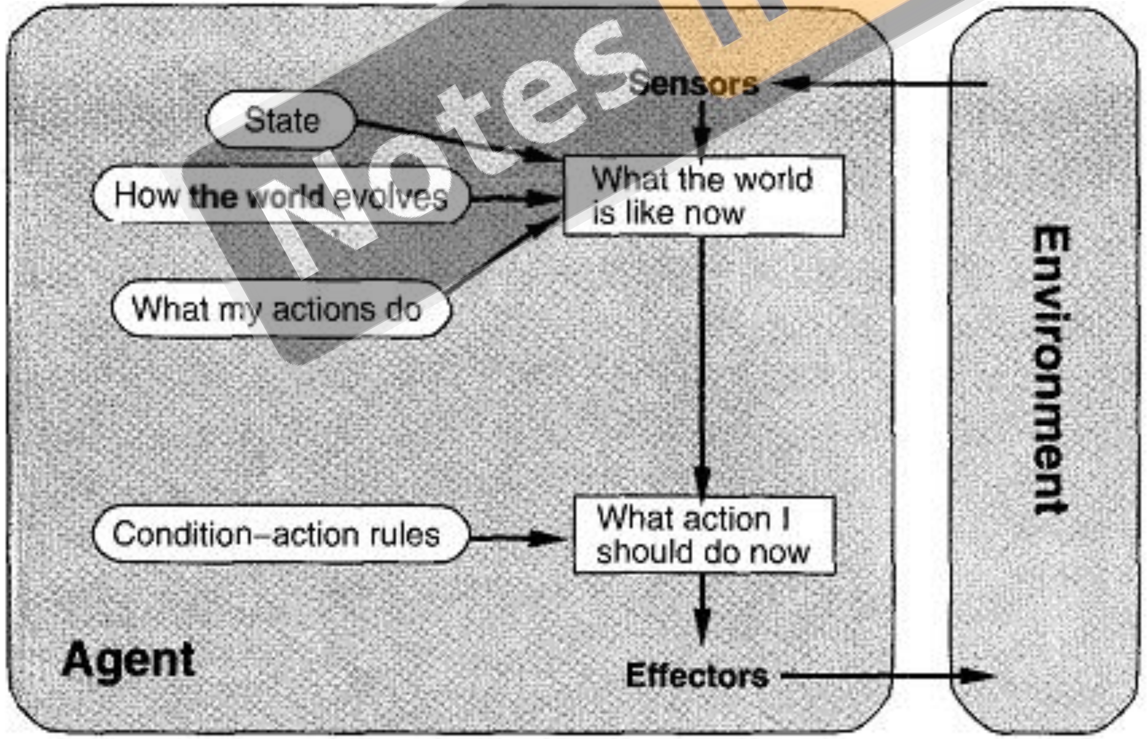
✓ This agent function works perfectly, only if the environment is fully observable.



Schematic diagram of a simple reflex agent.

## 2. Model-Based Reflex Agents

- These type of agents can handle partially observable environments. keeping track of the part of the world it can see now.
- It does this by keeping an internal state that depends on what it has seen before so it holds information on the unobserved aspects of the current state.
    - This time out mars Lander after picking up its first sample, it stores this in the internal state of the world around it so when it come across the second same sample it passes it by and saves space for other samples.
- It works by finding a rule whose condition matches the current situation.
-



### *3. Goal-based agents*

- It is an expansion of Model-based reflex agent.

- In life, in order to get things done we set goals for us to achieve, this pushes us to make the right decisions when we need to.

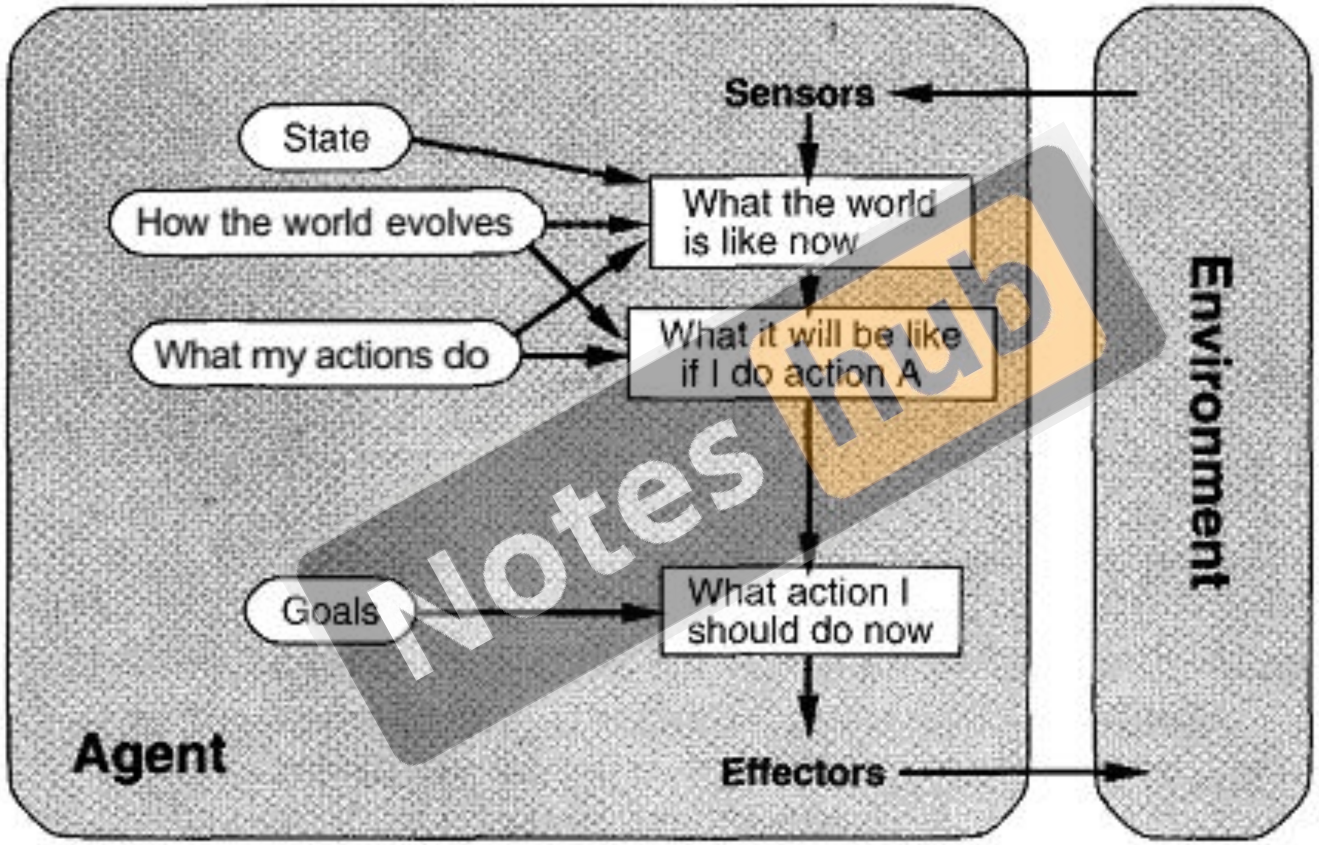
  - A simple example would be the shopping list; our goal is to pick up every thing on that list. This makes it easier to decide if you need to choose between milk and orange juice because you can only afford one. As milk is a goal on our shopping list and the orange juice is not we chose the milk.
  - An example of this IA class is any searching robot that has an initial location and wants to reach a destination.

- So in an intelligent agent having a set of goals with desirable situations are needed.

- The agent can use these goals with a set of actions and their predicted outcomes to see which action(s) achieve our goal(s).

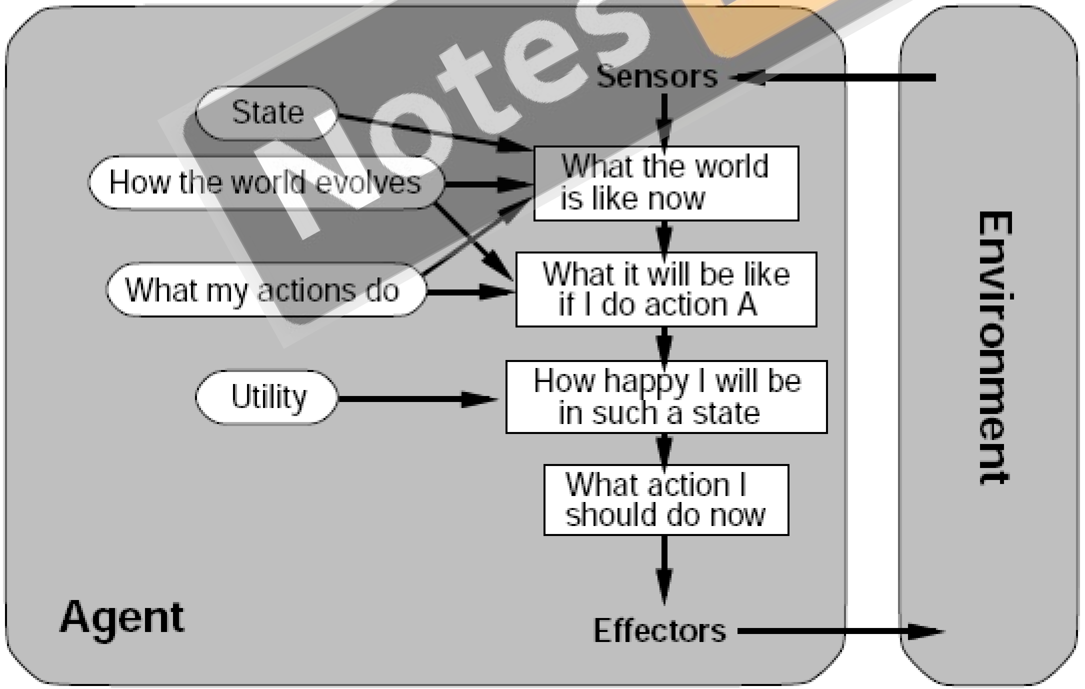
- Achieving the goals can take 1 action or many actions.

- Search and planning are two subfields in AI devoted to finding sequences of actions to achieve an agents goals.


Sensors
State
How the world evolves
What my actions do
What the world is like now
What it will be like if I do action A
Goals
What action I should do now
Agent
Effectors
Environment

## *4 Utility based agents*

- Goals are not really enough to generate high-quality behavior.
  - For example, there are many action sequences for a taxi to reach its destination i.e achieving the goal, but some are quicker, cheaper, safer, more reliable than other.
- Goals just provide a crude distinction between "happy" and "unhappy" states
- The word "utility" here refers to "the quality of being useful,“.
- Utility is therefore a function that maps a state  Or sequence of states, if we are measuring the utility of an agent over the long run.


Sensors
State
How the world evolves
What my actions do
Utility
What the world is like now
What it will be like if I do action A
How happy I will be in such a state
What action I should do now
Agent
Effectors
Environment

# Agent Environment

- An environment is everything in the world which surrounds the agent, but it is not a part of an agent itself.
- The environment is where agent lives, operate and provide the agent with something to sense and act upon it.

***As per Russell and Norvig, an environment can have various features from agent point of view:***

**1. Accessible vs Inaccessible:**
**2. Static vs Dynamic**
**3. Discrete vs Continuous**
**4.Deterministic vs non-Deterministic**
**5.Episodic vs non-episodic**

# General Problem Solver

The **General Problem Solver** (**GPS**) was an AI program proposed by *Herbert Simon, J.C. Shaw,* and *Allen Newell*.

- It was the first useful computer program that came into existence in the AI world. The goal was to make it work as a universal problem-solving machine.
- GPS was the first program that was intended to solve any general problem. GPS was supposed to solve all the problems using the same base algorithm for every problem.
- To program the GPS, the authors created a new language called **Information Processing Language** (**IPL**).

- The basic premise is to express any problem with a set of well-formed formulas. These formulas would be a part of a directed graph with multiple sources and sinks. In a graph, the source refers to the starting node and the sink refers to the ending node.
- In GPS, the source refers to axioms and the sink refers to the conclusions.

- ➢ Even though GPS was intended to be a general purpose, it could only solve well-defined problems, such as proving mathematical theorems in geometry and logic. It could also solve word puzzles and play chess.

## Solving a problem with GPS

1. The first step is to define the goals. Let's say our goal is to get some milk from the grocery store.
2. The next step is to define the preconditions. These preconditions are in reference to the goals. To get milk from the grocery store, we need to have a mode of transportation and the grocery store should have milk available.
3. After this, we need to define the operators. If my mode of transportation is a car and if the car is low on fuel, then we need to ensure that we can pay the fueling station. We need to ensure that you can pay for the milk at the store.

An operator takes care of the conditions and everything that affects them. It consists of actions, preconditions, and the changes resulting from taking actions. In this case, the action is giving money to the grocery store.

# Problem spaces and search / Problem state space search

- Any task to be done or goal to be achieved.
- Before, a solution can be found, the prime condition is that the problem must be very precisely defined.
- So, to solve a particular problem, we need to follow these things
    1. Define the problem precisely. This must include precise specifications of what the initial situations will be as well as final situations constitute acceptable solutions to the problem.
    2. Analyze the problem.
    3. Defining essential features – which can have an impact on accuracy of problem solving techniques.
    4. Defining knowledge to solve a particular problem.
    5. Applying best possible technique to solve it.

**The most common methods of problem representation in AI are**

- **State space representation**
- **Problem reduction.**

# Defining the Problem into State Space

*A set of all possible states for a given problem is known as the state space of the problem*

A problem should be defined in state space, which will include

- Initial state – Starting position
- Legal possible Solutions
- Goal state
- Rules applicable

But it tough to state all the rules correctly and will cause some storage problem also.

***Example***

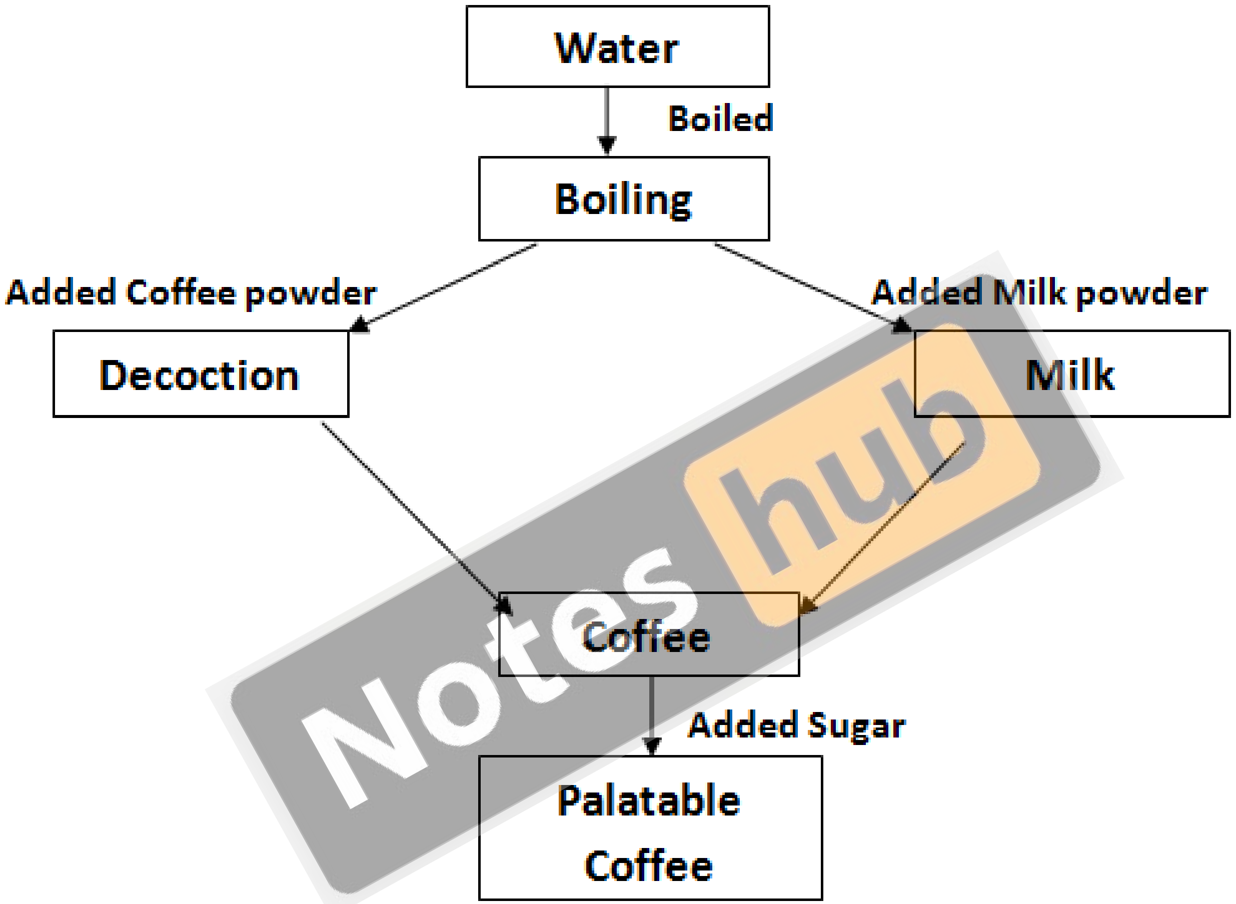
Suppose you are asked to make a cup of coffee.

What will you do? You will verify whether the necessary ingredients like instant coffee powder, milk powder, sugar, kettle, stove etc. are available.

If so, you will follow the following steps:

1. Boil necessary water in the kettle.
2. Take some of the boiled water in a cup add necessary amount of instant coffee powder to make decoction.
3. Add milk powder to the remaining boiling water to make milk.
4. Mix decoction and milk.
5. Add sufficient quantity of sugar to your taste and the coffee is ready.

Now think. You started with the ingredients (initial state), followed a sequence of steps (called states) and at last had a cup of coffee (goal state). You added only needed amount of coffee powder, milk powder and sugar (operators).



*State space representation of coffee making*

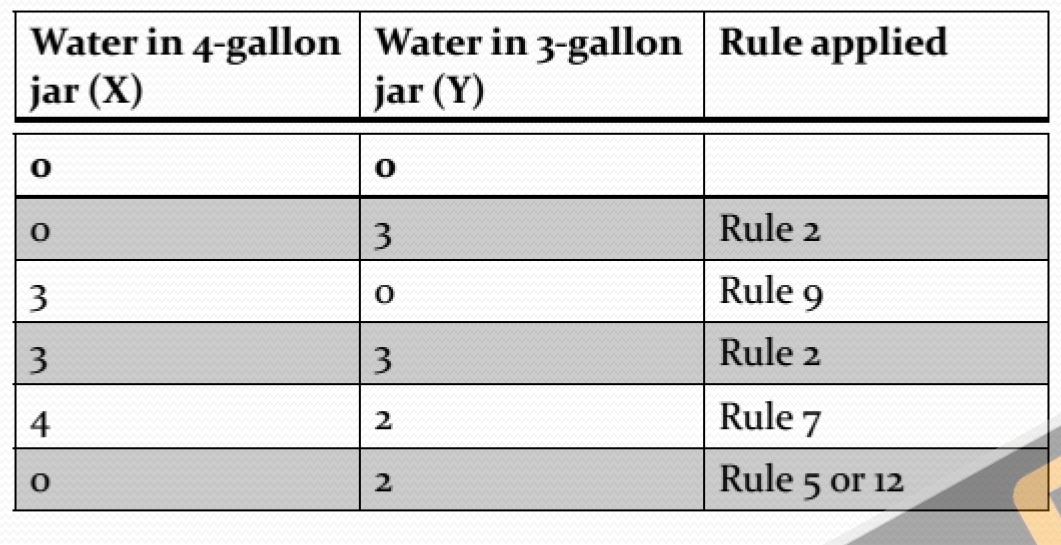
## Water Jug Problem

### Problem Formulation:

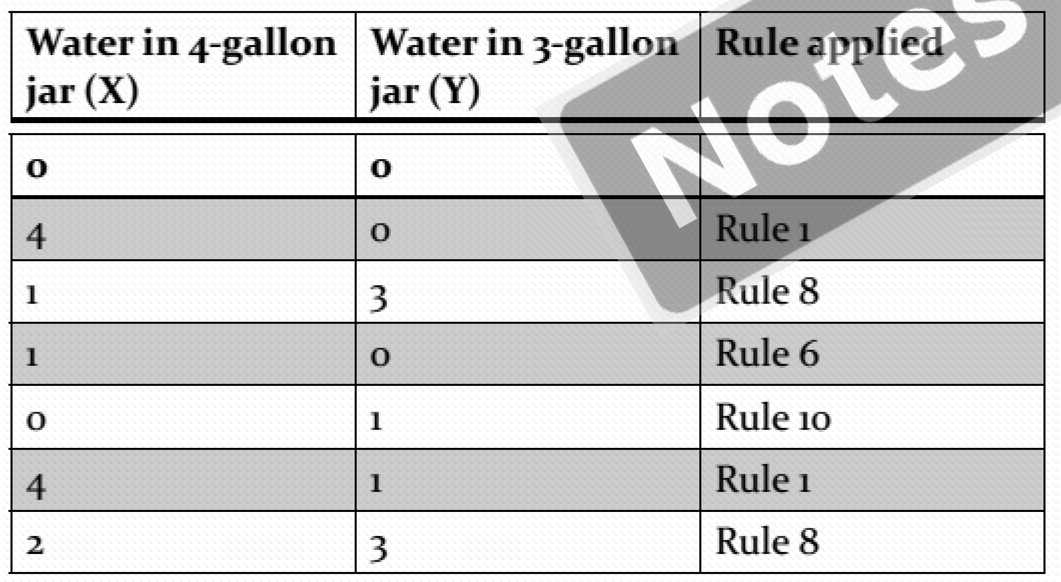
Each state= ordered pair {X,Y}
Where,
X= amount of water contained in Jar 1 (4 gallon capacity) at any time.
Y=amount of water contained in Jar 2 (3 gallon capacity) at any time.
Initial state: {0,0}
Final state: { P, 2}, Where P is any amount of water.

### Production Rules:

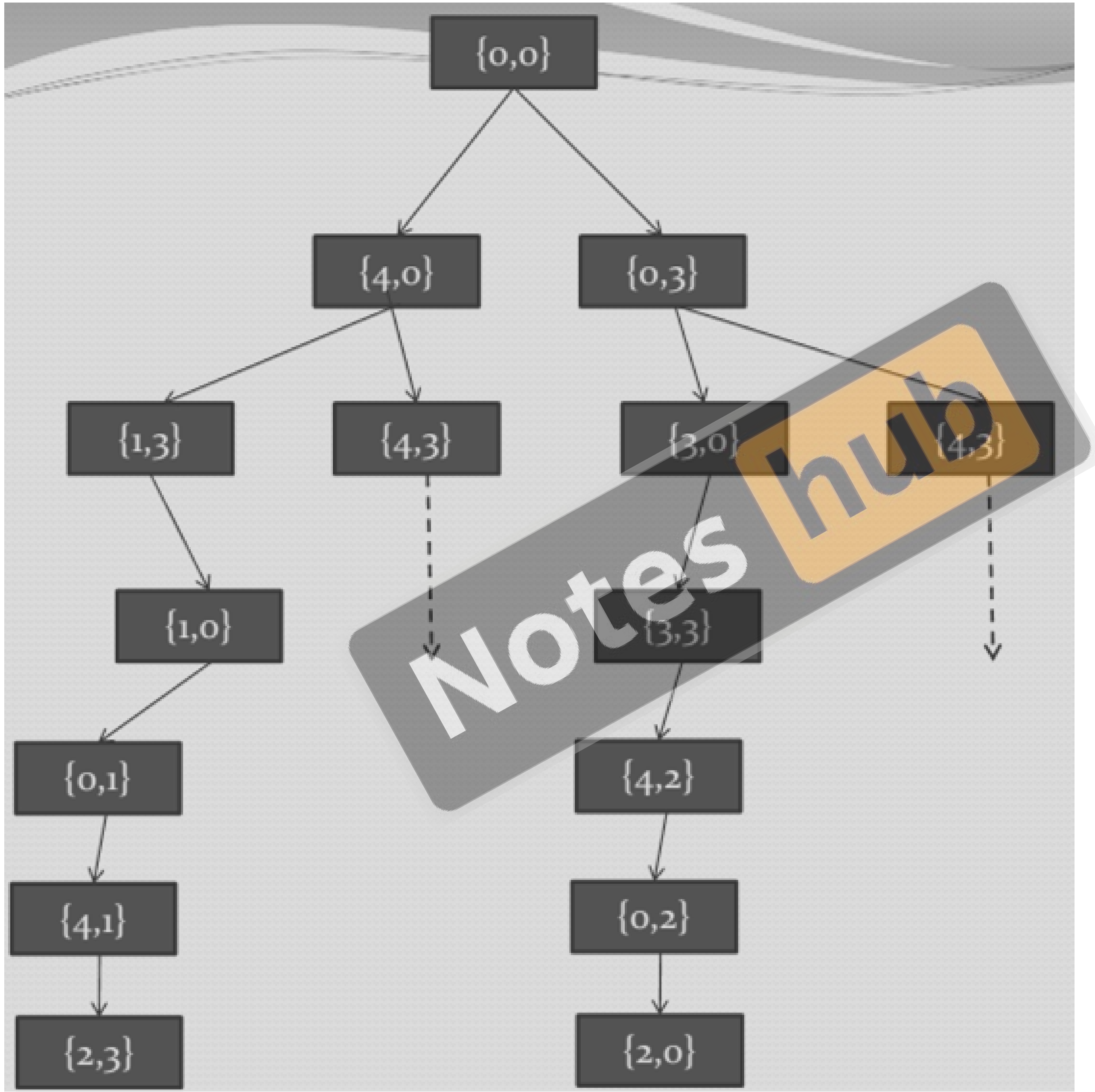
RULE 1: (X,Y)→(4,Y) [fill 4 gallon jug. Applicable if X<4]
RULE 2: (X,Y)→(X,3) [fill 3 gallon jug. Applicable if Y<3]
RULE 3: (X,Y)→(X-X1,Y) [pour some water out of 4 gallons jar]
RULE 4: (X,Y)→(X,Y-X1) [pour some water out of 3 gallons jar]
RULE 5: (X,Y)→(0,Y) [Empty 4 gallon jar]
RULE 6: (X,Y)→(X,0) [Empty 3 gallon jar]
RULE 7: (X,Y)→(4,Y-(4-X)) [Fill 4 gallon jar by pouring some water from 3 gallon jar]
RULE 8: (X,Y)→(X-(3-Y),3) [Fill 3 gallon jar by pouring some water from 4 gallon jar]
RULE 9: (X,Y)→(X+Y,0) [Empty 3 gallon jar by pouring all its water into 4 gallon jar]
RULE 10: (X,Y)→(0,X+Y) [Empty 4 gallon jar by pouring all its water into 3 gallon jar]
RULE 11: (0,2)→(2,0) [Pour 2 gallon from 3 gallon jar into 4 gallon]
RULE 12: (2,y)→(0,y) [empty the 2 gallons in 3 gallon jar on the ground]



| Water in 4-gallon jar (X) | Water in 3-gallon jar (Y) | Rule applied |
|---|---|---|
| 0 | 0 | |
| 0 | 3 | Rule 2 |
| 3 | 0 | Rule 9 |
| 3 | 3 | Rule 2 |
| 4 | 2 | Rule 7 |
| 0 | 2 | Rule 5 or 12 |



| Water in 4-gallon jar (X) | Water in 3-gallon jar (Y) | Rule applied |
|---|---|---|
| 0 | 0 | |
| 4 | 0 | Rule 1 |
| 1 | 3 | Rule 8 |
| 1 | 0 | Rule 6 |
| 0 | 1 | Rule 10 |
| 4 | 1 | Rule 1 |
| 2 | 3 | Rule 8 |


{0,0}
{4,0}
{0,3}
{1,3}
{4,3}
{3,0}
{4,3}
{1,0}
{3,3}
{0,1}
{4,2}
{4,1}
{0,2}
{2,3}
{2,0}

# Production Systems

*A structure that helps in describing and performing search in order to solve a problem is known as production system.*

A production system consists of :---

1. A set of rules , each consisting of left side ( a pattern) that determines the applicability of the rule and a right side that describes the operation to be performed. If the rule applied.

   These rules are called production rules or simply productions.

   Each rule is written in the form $C_i -> A_i$

   where

$C_i$ is the conditional part and $A_i$ is the action part.

A production system consists of a list of such rules----

$C_1 -> A_1, C_2 -> A_2, \dots C_m -> A_m$

2. One or more knowledge/ database that contain whatever information is appropriate for the particular task.

3. A control strategy that specifies the order in which the rule will be compared to the database and way of resolving the conflicts that arise when several rules match at once.
4. A rule applier.

One major issue to be sorted out in a production system is conflict resolution. Some of the conflict strategies are:--

- **Perform the first:** in this method the system chooses the first rule that matches.
- **Sequencing Techniques:** adopt the rules in the sequence they are.
- **Perform the most specific:** if there are two matching rules and one is more specific than the other, activate the most specific rule.
- **Most recent policy:** it is generally believed that a newly added rule is more knowledgeable than existing ones. Hence, if a system is adopting this method, it should fire the recent one.

  However, there is slight overhead associated with this strategy in that the system has to keep track of which rule came in at what time , which rules were modified etc.

# Control Strategy

Control strategy is a strategy by which we come to know which rule is to be applied next during the process of reaching  for a solution to a problem.
Control strategy help us to overcome the abnormal situations, when there are more than one rules or fewer than one rule will have its left sides match the current state.

A good control strategy must have the following requirements(characteristics).

1. ***It should cause motion:*** a control strategy should always cause motion as only such control strategy can lead to solution.
   if there is no motion that means there is no change of state and if state is not changed, then we will never proceed from our initial state. So reaching to a goal state will become a dream or impossible.

For example:  in water jug problem, if we choose first rule every time i.e fill the 4-liter jug full then we would never solve the problem. we would continue indefinitely filling the 4-liter jug with water.

**So a control strategy should always cause motion.**

1. ***It should be Systematic:*** A control strategy that causes motion but is not systematic is also not good. As in this type of strategy one can reach to a solution but it may pass from one state many times or may explore a particular sequence of operators several times unnecessarily.

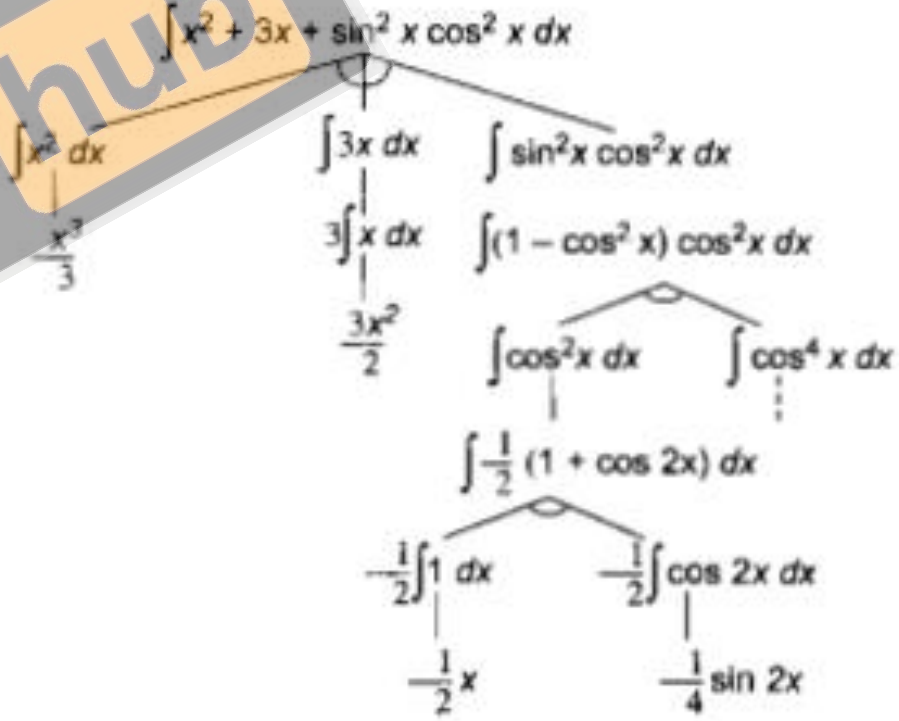
# Problem Characteristics

❖ **Is problem decomposable or not?**

If problem can be solved immediately then its ok, otherwise if decomposable then decompose it into simpler one, else apply some hard technique.

**Symbolic Integration**

$\int (x^2 + 3x + Sin^2x.Cos^2x\,)dx$



### ❖ Is problem undone or not?

- Chess → cannot undone (irrecoverable)
- Puzzle → can be undone (recoverable)

### ❖ Is universe predictable: solutions or moves predictable in advance or not?

In case of card game we cannot predict what is in opponents hand and what would be his reaction after our show. We can move according to our choice.

- Some problems are uncertain and undone, for them planning leads to high cost and time.
- Controlling a robot arm □ uncertain
- 8 Puzzle □ certain

### ❖ Is good solution a solution or a way to the solution?

Each state opted for solving a problem must have interaction with next or previous states

### ❖ What is the role of the knowledge?

- How much knowledge is required like chess requires only knowledge of legal and possible moves
- Which newspaper supports republicans or democrats requires high knowledge.

❖ **If problem requires interaction or not?**

- How much interaction between user and a system is required?
- Interaction can be for taking input, giving output, taking or giving instructions, updating knowledge base.

❖ **Is a good solution absolute or relative?**

- If we know initial state, goal state to be achieved and instructions to be followed then solution would be absolute else relative.

❖ **To which classification a problem belongs:**

Like Medical or Mechanical.

# Characteristics of a Production System

There are mainly four characteristics of the production system in AI that is simplicity, modifiability, modularity, and knowledge-intensive.

## Simplicity

The production rule in AI is in the form of an ‘IF-THEN’ statement. Every rule in the production system has a unique structure. It helps represent knowledge and reasoning in the simplest way possible to solve real-world problems. Also, it helps improve the readability and understanding of the production rules.

## Modularity

The modularity of a production rule helps in its incremental improvement as the production rule can be in discrete parts.

The addition or deletion of single information will not have a major effect on the output. Modularity helps enhance the performance of the production system by adjusting the parameters of the rules.

## Modifiability

The feature of modifiability helps alter the rules as per requirements.
Initially, the skeletal form of the production system is created.
We then gather the requirements and make changes in the raw structure of the production system.
This helps in the iterative improvement of the production system.

## Knowledge-intensive

Production systems contain knowledge in the form of a human spoken language, i.e., English. It is not built using any programming languages.
The knowledge is represented in plain English sentences. Production rules help make productive conclusions from these sentences.

# Production Classes

- A ***monotonic production system*** is a system in which the application of rule never prevents the later application of another rule that could also have been applied at the time that the first rule was selected.

- A ***nonmonotonic production system*** is one in which this is not true.

- A ***partially commutative production system*** is a system in with the property that if the application of particular sequence of rules transforms state x into state y, then any permutation of those rules that is allowable also transform state x in to state y.

- A ***commutative production system*** is a production system that is both monotonic and partially commutative.

# Relationship b/w problems and production systems

Ignorable problems; where creating new thins rather than changing old once

Change occur but can be reversed and in which order of operation is not critical

| | Monotonic | Nonmonotonic |
|---|---|---|
| **Partially Commutative** | Theorem Proving | Robot Navigation, 8-puzzle |
| **Not Partially Commutative** | Chemical synthesis | Bridge, Chess |

where creating new things by changing old once

Reverse not possible and order matter.