

## AI Experiment 1

DATE:

Koleena Shah

60004210243

Aim : To select a problem statement related to AI

- (1) Identify the problem
- (2) PEAS description
- (3) Problem Formulation

Theory :

Problem : Autonomous Vehicle Navigation

The problem of autonomous vehicle navigation involves creating AI systems that enable vehicles to autonomously navigate through real world environments. This entails developing algorithms that allow vehicle to perceive their surroundings using sensors like cameras, LiDAR, radar & GPS, make decisions based on this information to drive safely, obey traffic rules, avoid obstacles & pedestrians & efficiently reach their destinations.

The challenge lies in creating robust systems capable of handling diverse road conditions, unpredictable scenarios, & ensuring passenger safety throughout the journey.

PEAS description

Performance Measure

The successful navigation of a vehicle from one point to another while obeying traffic rules, avoiding accidents & reaching destination safely.

## A. Environment TA

~~data present~~

### Environment

Roads, traffic signals, pedestrians, other vehicles, weather conditions, road obstacles

to monitor environment according to required criteria

### Actuators

Steering, acceleration, braking mechanism, sensors, GPS

### Sensors

Cameras, LiDAR, radar, motion sensors & other environmental sensors to monitor & avoid

### Problem Formulation

(1) Initial State

(2) Actions

(3) Transition Model

(4) Goal Test

(5) Path Cost

**Conclusion:** Thus, we understood details & importance

## AI Experiment, 2

Kleena Shah,

1(2018) Adm no. 60004210243.

C'32

Topic: Implementation of search algorithm with example BFS

Aim: To implement uninformed search algorithms

(1) BFS using queue data structure

(2) DFS

(3) DFID (using stack with frontier tree)

Theory: PFS had a significant shortcoming (blind).

Using informed search algorithms, blind search algorithms or brute force algorithm are used to find solutions in search space without any prior information or heuristics.

They explore the search space systematically & do not consider the quality of the paths or the states they explore.从而它们只关注于搜索空间中的所有可能状态。

Three commonly used uninformed search algorithms are as follows:

(1) BFS (using queue data structure, problem solving approach)

(2) DFS (using stack data structure, problem solving approach)

(3) DFID (using stack, frontier tree, problem solving approach)

### Breadth First Search

It explores a search tree by expanding all the child nodes at a given depth before moving to the next level.

## Depth First Search (DFS)

It explores the search tree by selecting a child node & diving as deeply as possible before backtracking.

Usually implemented using stack

## Depth First Iterative Deepening (DFID)

DFID combines the benefits of both BFS & DFS.

It performs a series of DFS searches with increasing depth limits starting from depth of 1 & incrementally increasing the depth limit until the goal state is found.

**Conclusion :** The choice of which uninformed search to use depends on the specific characteristics of the problem.

BFS guarantees optimality but may consume more memory, making it suitable for problems with limited search space

DFS is memory efficient, but may not guarantee optimality, making it useful for deep search space

DFID strikes a balance by combining memory efficiency & with the guarantee of finding the shortest path

## AI Experiment 3

Kreema Shah

60004210243

C32 17/07/23

Aim : To implement A\*, informed search algorithm.

Theory :

Informed search algorithms, heuristic search algorithms are designed to optimize the search process by using problem specific information or heuristics.

Unlike, uninformed search which explores the search space blindly, informed search algorithm aim to prioritize promising paths that are more likely to lead to a solution.

They do so by assigning a cost or value to each essential potential path based on a heuristic function which estimates the desirability of a particular state or node.

The key components are :

- (1) Heuristic Function
- (2) Cost Function
- (3) Open set & Closed set
- (4) Priority queue

#### Heuristic Function

It is a domain specific estimate that assigns a value to each state or node in the search space.

Info about

Cost function

Represents the total cost incurred so far to reach a specific state from initial state

Open & Closed Set

Data structure used to keep track of states that need to be explored & have already been explored.

Priority Queue

It employs this to select the next state to explore

Conclusion: A\* efficiently solves complex problems by intelligently navigating through search spaces.

A\* is built using following features of search algorithm of A\*.

Searches in direction of goal and explores states

in direction of goal, it can avoid exploring states

which are not promising or which are not associated

with high cost of reaching goal state.

It also avoids states which are not promising

and which are not associated with high cost of reaching goal state.

The benefit of the above two

swings utilization of

information available

and also reduces number of states

explored during search process.

## AI Experiment 4

DATE:

Kreena Shah

6000420243

13210210

minimise (c)

Aim : To implement local search algorithm : hill climbing

Theory :

Hill climbing is a local search algorithm in AI to find the best possible solution to a problem.

It is named after the idea that it climbs the hills of search space to find the most optimal solutions but it can have limitations & may not be always find the global optimum too many times.

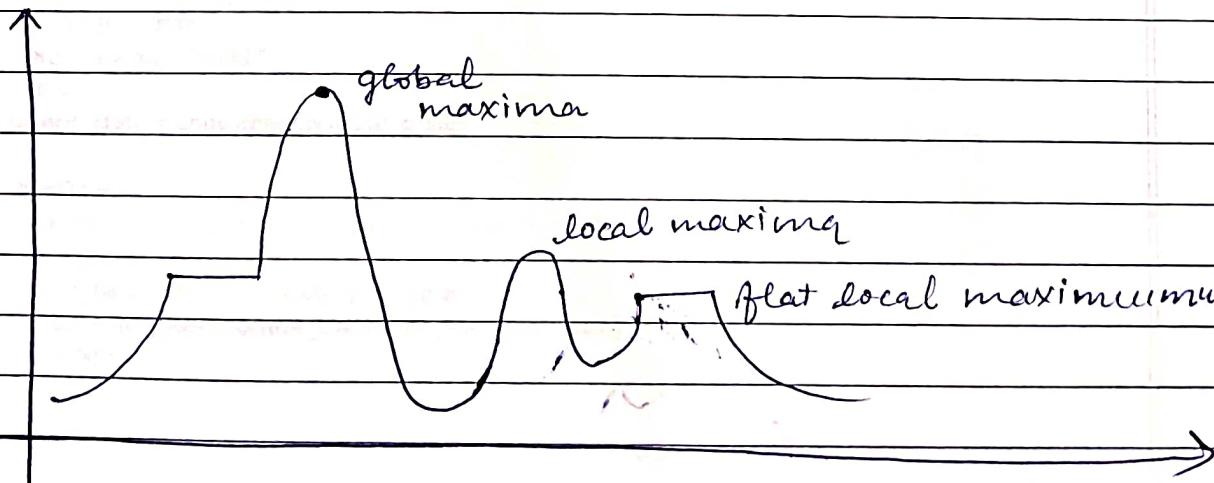
Features of Hill climbing

Generate & Test Variant

Greedy Approach

No backtracking

State Space diagram



## Hill Climbing

Advantages:

- (1) Simplicity
- (2) Efficiency

Disadvantages:

- (1) Limited Exploration
- (2) Local Optimum
- (3) Dependency on initial state
- (4) It does not maintain a history of visited states

**Conclusion:** Hill Climbing can be beneficial for unstructured, non-monotonously increasing paths in search space but it cannot be relied on completely as it may get stuck at local minima.

## Variants of Hill Climbing

1. Steepest Ascent Hill Climbing

2. Randomized Hill Climbing

3. Simulated Annealing

4. Non-greedy hill climbing

5. Simulated Annealing

6. Genetic Algorithm

7. Ant Colony Optimization

8. Particle Swarm Optimization

9. Tabu Search

DATE:

## AI Experiment 5

Kaleena Shah

60004210243

Aim : To implement genetic algorithm to solve optimization problems

Theory :

A genetic algorithm is a search heuristic reflecting the process of survival of the fittest for producing the next generation.

There are primarily 5 phases:

- (1) Initialisation
- (2) Fitness Assignment
- (3) Selection
- (4) Reproduction
  - Crossover
  - Mutation
- (5) loop through 1st 4 steps

Initialisation

A set of individuals called as population each being a solution to the given problem.

Fitness function

To determine the ability of an individual to compete with others

## Genetic Algorithm

### Selection

The selection of individuals for reproductions of the next generation

→ Selection of individuals with higher fitness

### Reproduction

The creation of children in next generation is done

- Crossover

- Mutation

→ Selection of individuals with higher fitness

**Conclusion:** Genetic algorithm can be used to find solution using local moves renewing population with the best solutions

→ Selection of individuals with higher fitness

## AI Experiment 6

Kreema Shah

60004210243

Institutional number : C'32

Date : 10/10/2023

Aim : To study & implement Perceptron learning  
Algorithm.

## Theory :

Perceptron learning is a fundamental concept in artificial machine learning & artificial intelligence particularly neural network

## Structure :

It consists of input nodes associated with weights & an output node.

The inputs & weights are multiplied & the weighted sum is passed through an activation function.

The output is a binary decision based on whether the weighted sum exceeds a certain threshold which is usually a step function.

The above process involves the following steps, mentioned below

(1) Initialization

(2) Calculation for each input

(3) Error calculation & weight input

$$\Delta w = c \text{ (desired - output)} i$$

$$w_{i+1} = \Delta w + w_i$$

(4) Repeat

## 2) Perceptron TA

Date dictated

**Conclusion :** Perceptron learning is a straightforward method for training a basic neural network model which while it has limitations, it played crucial role in history of neural network.

: present

• It gives a backpropagation or gradient descent method  
• Gradient descent is a general optimization technique  
• Significant for early applications

• Applications

• Image, after LeCun's work in 1980s  
• Handwriting recognition

• Recognition of handwritten digits is still a difficult problem  
• Ambiguity, variability and difficulty in reading the digits  
• Consideration of local variations around a structural unit  
• Involves local neighborhood in choosing initial weights  
• Weight update

• Iterating, take initial lab with random starting point & add

• weight

• weight

• weight

• Training and updating (s)

• Input values &amp; target values (s)

• ( $x_1, x_2, \dots, x_n$ ) =  $s$ •  $(t_1, t_2, \dots, t_n) = s$ 

• Output

## AI Experiment 7

DATE:

Kareena Shah

60004210243

C'32

Aim : Program to implement family tree in Prolog

Theory :

Prolog is a logic programming language.

It has an important role in artificial intelligence.

Unlike many other programming languages, Prolog is intended primarily as a declarative programming language.

In prolog, logic is expressed as relations (called as facts & rules)

Formulation is carried out by running a query over these relations

Key features

- (1) Unification
- (2) Backtracking
- (3) Recursion

Advantages

- (1) Easy to build database
- (2) Pattern matching is easy
- (3) It has built-in list handling

Disadvantages

- (1) LISP dominates over prolog with respect to I/O features
- (2) Sometimes I/O is not easy.

DATE:

## # for simplicity TA

(e) Conclusion

Conclusion : Thus, we studied & implemented protoprotocol.

Protocol: our first attempt to implement the protocol is successful.

Protocol

Protocol receives new signal in the order

arrangement: Initiation of state, transmission of data, PDU

Initiation

or end-of-reception acknowledgement -> data transfer

receiving acknowledgement or transmission of data

Transmission

> data transfer or acknowledgement of data transfer

(continues)

Protocol

Protocol receives acknowledgement of data transfer

acknowledgment

acknowledgment

acknowledgment (1)

acknowledgment (2)

acknowledgment (3)

acknowledgment (4)

acknowledgment (5)

acknowledgment (6)

acknowledgment (7)

acknowledgment (8)

acknowledgment (9)

acknowledgment (10)

acknowledgment (11)

acknowledgment (12)

acknowledgment (13)

acknowledgment (14)

acknowledgment (15)

acknowledgment (16)

acknowledgment (17)

acknowledgment (18)

acknowledgment (19)

acknowledgment (20)

acknowledgment (21)

acknowledgment (22)

acknowledgment (23)

acknowledgment (24)

acknowledgment (25)

acknowledgment (26)

acknowledgment (27)

acknowledgment (28)

acknowledgment (29)

acknowledgment (30)

acknowledgment (31)

acknowledgment (32)

acknowledgment (33)

acknowledgment (34)

acknowledgment (35)

acknowledgment (36)

acknowledgment (37)

acknowledgment (38)

acknowledgment (39)

acknowledgment (40)

acknowledgment (41)

acknowledgment (42)

acknowledgment (43)

acknowledgment (44)

acknowledgment (45)

acknowledgment (46)

acknowledgment (47)

acknowledgment (48)

acknowledgment (49)

acknowledgment (50)

acknowledgment (51)

acknowledgment (52)

acknowledgment (53)

acknowledgment (54)

acknowledgment (55)

acknowledgment (56)

acknowledgment (57)

acknowledgment (58)

acknowledgment (59)

acknowledgment (60)

acknowledgment (61)

acknowledgment (62)

acknowledgment (63)

acknowledgment (64)

acknowledgment (65)

acknowledgment (66)

acknowledgment (67)

acknowledgment (68)

acknowledgment (69)

acknowledgment (70)

acknowledgment (71)

acknowledgment (72)

acknowledgment (73)

acknowledgment (74)

acknowledgment (75)

acknowledgment (76)

acknowledgment (77)

acknowledgment (78)

acknowledgment (79)

acknowledgment (80)

acknowledgment (81)

acknowledgment (82)

acknowledgment (83)

acknowledgment (84)

acknowledgment (85)

acknowledgment (86)

acknowledgment (87)

acknowledgment (88)

acknowledgment (89)

acknowledgment (90)

acknowledgment (91)

acknowledgment (92)

acknowledgment (93)

acknowledgment (94)

acknowledgment (95)

acknowledgment (96)

acknowledgment (97)

acknowledgment (98)

acknowledgment (99)

acknowledgment (100)

acknowledgment (101)

acknowledgment (102)

acknowledgment (103)

acknowledgment (104)

acknowledgment (105)

acknowledgment (106)

acknowledgment (107)

acknowledgment (108)

acknowledgment (109)

acknowledgment (110)

acknowledgment (111)

acknowledgment (112)

acknowledgment (113)

acknowledgment (114)

acknowledgment (115)

acknowledgment (116)

acknowledgment (117)

acknowledgment (118)

acknowledgment (119)

acknowledgment (120)

acknowledgment (121)

acknowledgment (122)

acknowledgment (123)

acknowledgment (124)

acknowledgment (125)

acknowledgment (126)

acknowledgment (127)

acknowledgment (128)

acknowledgment (129)

acknowledgment (130)

acknowledgment (131)

acknowledgment (132)

acknowledgment (133)

acknowledgment (134)

acknowledgment (135)

acknowledgment (136)

acknowledgment (137)

acknowledgment (138)

acknowledgment (139)

acknowledgment (140)

acknowledgment (141)

acknowledgment (142)

acknowledgment (143)

acknowledgment (144)

acknowledgment (145)

acknowledgment (146)

acknowledgment (147)

acknowledgment (148)

acknowledgment (149)

acknowledgment (150)

acknowledgment (151)

acknowledgment (152)

acknowledgment (153)

acknowledgment (154)

acknowledgment (155)

acknowledgment (156)

acknowledgment (157)

acknowledgment (158)

acknowledgment (159)

acknowledgment (160)

acknowledgment (161)

acknowledgment (162)

acknowledgment (163)

acknowledgment (164)

acknowledgment (165)

acknowledgment (166)

acknowledgment (167)

acknowledgment (168)

acknowledgment (169)

acknowledgment (170)

acknowledgment (171)

acknowledgment (172)

acknowledgment (173)

acknowledgment (174)

acknowledgment (175)

acknowledgment (176)

acknowledgment (177)

acknowledgment (178)

acknowledgment (179)

acknowledgment (180)

acknowledgment (181)

acknowledgment (182)

acknowledgment (183)

acknowledgment (184)

acknowledgment (185)

acknowledgment (186)

acknowledgment (187)

acknowledgment (188)

acknowledgment (189)

acknowledgment (190)

acknowledgment (191)

acknowledgment (192)

acknowledgment (193)

acknowledgment (194)

acknowledgment (195)

acknowledgment (196)

acknowledgment (197)

acknowledgment (198)

acknowledgment (199)

acknowledgment (200)

acknowledgment (201)

acknowledgment (202)

acknowledgment (203)

acknowledgment (204)

acknowledgment (205)

acknowledgment (206)

acknowledgment (207)

acknowledgment (208)

acknowledgment (209)

acknowledgment (210)

acknowledgment (211)

acknowledgment (212)

acknowledgment (213)

acknowledgment (214)

acknowledgment (215)

acknowledgment (216)

acknowledgment (217)

acknowledgment (218)

acknowledgment (219)

acknowledgment (220)

acknowledgment (221)

acknowledgment (222)

acknowledgment (223)

acknowledgment (224)

acknowledgment (225)

acknowledgment (226)

acknowledgment (227)

acknowledgment (228)

acknowledgment (229)

acknowledgment (230)

acknowledgment (231)

acknowledgment (232)

acknowledgment (233)

acknowledgment (234)

acknowledgment (235)

acknowledgment (236)

acknowledgment (237)

acknowledgment (238)

acknowledgment (239)

acknowledgment (240)

acknowledgment (241)

acknowledgment (242)

acknowledgment (243)

acknowledgment (244)

acknowledgment (245)

acknowledgment (246)

acknowledgment (247)

acknowledgment (248)

acknowledgment (249)

acknowledgment (250)

acknowledgment (251)

acknowledgment (252)

acknowledgment (253)

# AI Experiment, 8

DATE:

Kareena Shah

60004210243

C'32

Aim : Implementation on any AI problem

- (1) Wumpus World
- (2) Tic - tac - toe
- (3) 8 - queens

Theory :

## 8 - Queens Problem

The 8-queen problem involves placing eight queens on an 8x8 chessboard in a way that no two queens threaten each other

Implementation Approach

### (1) State Representation

Represent the chessboard 8 queens position

### (2) Constraint Satisfaction

Use constraint programming / backtracking algorithms to place queens on the board while satisfying constraints

### (3) Search Techniques

Employ various search strategies, such as recursive backtracking, to explore possible queen placements efficiently without violating constraints

DATE:

## Q. Brute-force TA

introduction

Conclusion: Thus, we implemented 8-queens problem

using

Brute-force TA given us wait-free algorithm? write

1 6

Intelligent (A)

not-optimal (S)

optimal (E)

\*

→ pseudocode

## Wait-free algorithm (D) - 8

main idea: this strategy contains wait-free reading & writing

Trivial: crossing out one coordinate in boardsize 8x8

result: 1702

Intelligent wait-free algorithm?

wait-free reads writes (W)

wait-free writing & boardsize and traversing

wait-free reads (R)

Intelligent wait-free algorithm & wait-free reading & writing

## Combination algorithm (S)

Intelligent wait-free, intelligent writing and reading (W)

Intelligent wait-free reading and writing of wait-free reading and writing

## AI Experiment 9

DATE:

Koleena Shah

60004210243

C'32 sem 1 (1)

Platform: LMS

Aim : Identify, analyze, implement a planning problem / Rule based Expert system in a real world scenario.

Let's consider a real world scenario where a rule based expert system could be applied : Hospital Bed Management System

### Problem Identification

Hospital bed management involves efficiently allocating & managing available beds for patients based on their medical needs, urgency & the hospital's resources.

### Analysis

A hospital has various types of beds (ICU, general, pediatric) & a fluctuating no. of patients with different medical conditions requiring admission.

### Challenges

- (1) Bed Allocation
- (2) Optimization
- (3) Dynamic Environment

### Rule Based Expert System

- (1) Knowledge Base
- (2) Inference Engine
- (3) User Interface

## P. Hospital Bed Management

Ch 12: Hospital Bed Management

Rules & logic :

(1) Priority Rules

(2) Bed Availability

(3) Patient Conditions

(4) Dynamic Updates

Implementation :

(1) Data Collection : hospital bed utilization monitoring

(2) Rule Formulation

(3) System Development

(4) Testing & Validation

(5) Integration & Deployment

Outcome :

An efficient rule based expert system for hospital bed management would :

Optimize bed allocation based on patient needs & available resources

Reduce wait times for critical cases

Provide a systematic approach for bed assignment, ensuring better patient care & resource utilization

Conclusion : Rule based expert systems contribute to improved patient outcomes & resource management in healthcare facilities

Kareena Shah  
60004210243

## Experiment 10

### Case Study - AI Application Research Paper Summary

Paper Title	Composer4Everyone: Automatic Music Generation with Audio Motif	An AI-Based Intelligent Music Composing Algorithm: Concord	Music Generation with AI Technology: Is It Possible?
Conference Name & Year	2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)	2013 International Conference on Advances in Technology and Engineering (ICATE)	2022 IEEE 5th International Conference on Electronics Technology (ICET)
Algorithm	The algorithm processes user input (humming, singing, speaking) and generates a 20-30 second piano piece. It employs expectation analysis to ensure alignment with user expectations and a rearranging system for style variations.	Concord is an object-oriented algorithm focusing on scales, chords, and note patterns. It balances musical rules and randomness to create coherent compositions, using the circle of fifths for harmony.	Reviews Biaxial-LSTM, DeepJ, and MuseGAN. These models are employed for polyphonic and pop music generation, with DeepJ focusing on style-specific generation and MuseGAN using GANs for pop music.
Dataset	The paper does not explicitly mention the dataset used for developing or training the algorithm.	No specific dataset mentioned; the algorithm is based on musical theory and object-oriented programming principles.	The paper does not detail specific datasets but reviews literature on various models and their applications in AI music generation.
Results	The application was tested with 300 participants in a listening test, showing high satisfaction with the quality and consistency of the generated music.	The paper does not provide empirical results or user testing outcomes, focusing more on the theoretical aspects of the algorithm.	The paper does not report specific results but provides an analysis of the models' capabilities and scenarios where they are applied.
Observations regarding performance	High user satisfaction, indicating effectiveness in generating music that aligns with user expectations. The system's interaction design and expectation analysis are key to its performance.	Concord's performance is not empirically tested but is theorized to produce unique and musically coherent pieces based on its algorithmic design.	The comparison of models highlights the strengths and limitations in AI music generation, emphasizing the need for further

			research in this area.
Conclusion	The app demonstrates the potential of AI in making music composition accessible to a broader audience, with promising results in user interaction and satisfaction.	Concord represents a novel approach in AI music composition, combining musical theory with computational methods, though practical applications and user experiences are yet to be explored.	The paper concludes that while AI has made significant strides in music generation, there is still a need for models that can better emulate human creativity and emotional expression.

Page No.	
Date	

01/11/2023

Name : Kireena Shah

Sapid : 60004210243

Batch : Comps C'32

Subject : AI Assignment 1

## (1) (i) Semantic Network

(a) A semantic network is a knowledge structure that depicts how concepts are related to one another & illustrates how they are interconnected.

(b) Semantic networks use artificial intelligence (AI) programming to mine data, connect concepts & call attention to relationships.

(c) In business, this capability can make customer service more proficient by providing better product search functionality.

(d) New technologies like Microsoft Office Graph use semantic networking to bring related concepts together.

## (ii) RDF &amp; OWL

- Resource Description Framework Web Ontology language is a framework for describing powerful ontology language & linking resources on web that extends RDF.

- It is primarily used for representing data in a structured & machine readable format. It is designed to model & reason about complex relationships & ontologies in a formal & expressive way.

- RDF uses triples to express statements / facts about creation of ontologies that

Resources

define classes, properties & relationships among data elements

- It provides a basic graph based data model for representing the information

It provides a rich set of logical constructs for defining complex concepts such as classes, individuals & properties & supports reasoning about these concepts

(2) (i) Ant Colony Optimization (ACO) is a nature inspired optimization algorithm based on the foraging behaviour of ants.

(ii) ACO is used to find solutions to complex combinatorial optimization problems

→ Inspiration : ACO is inspired by the way real ants find the shortest path between their nest & a food source by depositing pheromones. Ants follow paths with higher pheromone concentrations

→ Solution Construction : In ACO, artificial "ants" iteratively construct solutions by moving through a problem space. They make decisions based on both pheromone levels & a heuristic function that guides them toward promising solutions

→ Pheromone Update : After constructing a solution, ants update the pheromone levels on the paths they have traversed. Shorter & better solutions receive higher pheromone deposits.

Page No.	
Date	

- Exploration vs Exploitation : ACO balances exploration (searching for new solutions) & exploitation (reinforcing known good solutions) by adjusting the pheromone evaporation rate & the influence of pheromone & heuristic information in decision making
- Convergence : Over time, ACO tends to converge towards optimal or near optimal solutions as good solutions accumulate pheromone & become more attractive to ants

### (3) Unification Algorithm for Predicate Logic

- (1) The unification algorithm is used to find a substitution that makes two predicates equals in first order logic.
  - (2) It's crucial for reasoning & resolution in predicate logic.
  - (3) Unification is a process of finding a common instantiation of variables in two predicates to make them the same.
- Eg : Consider predicates  $P(x, y)$  &  $P(a, b)$ . To unify them, we want to find values for  $x$  &  $y$  such that  $P(x, y)$  becomes the same as  $P(a, b)$ .

#### Algorithm :

- (a) Start with an empty substitution : {}
- (b) Compare the arguments of the predicates
  - $x$  unifies with  $a$ , so  $\{x/a\}$  to the substitution.

Page No.	
Date	

- $y$  unifies with  $b$ , so add  $\{x/a, y/b\}$  to the substitution.
- (c) Apply the final substitution to the original predicates to get  $P(a,b)$  &  $P(a,b)$  which are same.

#### (4) Bayesian Belief Network {BBN}

- (a) A Bayesian Belief Network is a probabilistic graphical model used for representing & reasoning about uncertainty in a system.
- (b) It consists of nodes representing random variables & directed edges showing probabilistic dependencies between them.
- (c) Bayesian Belief Network are used in various applications such as decision support & machine learning.
- (d) They incorporate Bayesian probability theory to update beliefs as new evidence is obtained.

#### (5) Fuzzy Set & Operations

- (a) A fuzzy set is a mathematical concept that generalizes the idea of classical sets.
- (b) It allows elements to belong to a set to a degree, ranging from 0 (completely outside the set) to 1 (completely inside the set).
- (c) Fuzzy sets are used in fuzzy logic to handle uncertainty.

Fuzzy Set Operations :

Page No.		
Date		

### (a) Union

The union of two fuzzy sets  $A \cup B$  at each point represents the maximum degree to which an element belongs to either  $A$  or  $B$

### (b) Intersection

The intersection of two fuzzy sets  $A \cap B$  at each point represents the minimum degree to which an element belongs to both  $A$  &  $B$

### (c) Complement

The complement of a fuzzy set  $A$  represents the degree to which elements do not belong to  $A$ , often denoted as "not  $A$ "

# AI Assignment 2

DATE:

Kreena Shah

60004210243 (ii)

(1) Hierarchical Planning: In AI involves organizing complex planning problems into a hierarchy of tasks or actions, enabling more efficient & structured problem solving. It breaks down a large task into smaller, more manageable sub-tasks, creating a multi-level structure where higher level plans control & guide lower level plans.

## Key Components :

### (1) Abstraction levels

Higher level represent broader goals / tasks & lower levels details the specific actions required to achieve them.

### (2) Decomposition

The process involves decomposing high level goals into sub-goals. Each sub-goal can further decompose into more detailed actions until reaching a level where actions can be directly executed.

### (3) Information flow

Here, information flows in both ways in a hierarchy. Higher level plans provide guidance & constraints to lower level plans while feedback from lower level helps in refining higher level plans.

## e Learning@TA

### Information

#### (4) Efficiency

Hierarchical Planning can be significantly reduce computational complexity by focusing on specific part of problem at each level, leading to more efficient planning algorithms.

#### (2)

#### Goal Stack Planning (GSP)

GSP is a method used in artificial intelligence for automated planning. It involves representing a problem-solving process by breaking down complex goals into simpler subgoals to achieve a desired outcome.

It utilizes a stack-based structure to organize and execute these subgoals.

#### Example : Making Breakfast

##### (1) Goal : Make Pancakes

Subgoals :

- Gather ingredients
- Mix batter
- Heat skillet
- Cook pancakes
- Serve pancakes

##### Subgoal decomposition

- Serve Pancakes
- Prepare plates
- Place pancakes, add toppings
- Serve

- Cook Pancakes
  - Heat up traps on a skillet (S)
  - Pour batter on skillet (P)  $\rightarrow$  invertible (I)
  - Flip pancakes in constant time (D)  $\rightarrow$  Invertible (I)
  - Ensure proper cooking time in  $\approx$  1 minute (T)

**Execution Order:**  $S \rightarrow P \rightarrow D \rightarrow T$

The system executes each subgoal in a LIFO manner from the goal stack, working backward until the primary goal is achieved.

### → Goal Stack Planning vs Forward State Space Planning (FSSP)

- Operates by decomposing a high level goal into subgoals & solving them to achieve the primary objective.
- Utilizes a stack structure to organize & manage subgoals.
- Operates in depth first search manner, working backward from final goal through subgoal decomposition.

Represents the planning problem as a state space graph, exploring possible actions & states to reach goal.

Initial state, incrementally moving towards goal state.

(3)

Building an expert system has following phases

(1) Identification & Selection of Problem

The initial phase involves identifying problem domain suitable for an expert system.

This could be an area where expertise is required such as medical diagnosis, financial analysis or technical troubleshooting in various fields such as

(2) Knowledge Acquisition

This phase involves gathering knowledge from domain experts.

(3) Knowledge Representation

Here acquired knowledge is organized & integrated into systems.

(4) Inference Engine Development

The inference engine is the core component responsible for reasoning & making inference based on the knowledge stored.

(5) User Interface Design

It allows user to interact with expert system.

It is user-friendly

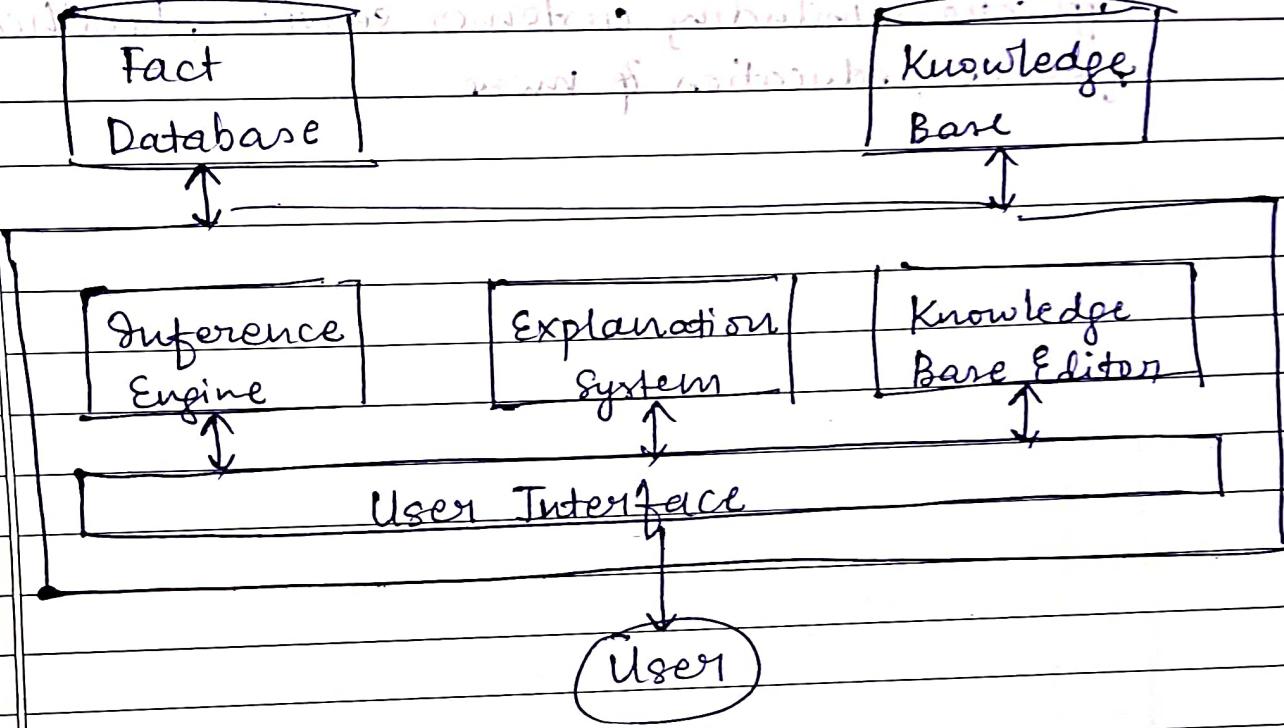
(6) Testing & Validation

Once expert system is developed, it undergoes testing & validation.

DATE:

## (7) Deployment & Maintenance

After successful testing, expert systems are deployed for real world use.



(4) We will discuss Natural Language Processing (NLP) which focuses on enabling machines to understand, interpret, generate & respond to human language in a way that is both meaningful & contextually relevant.

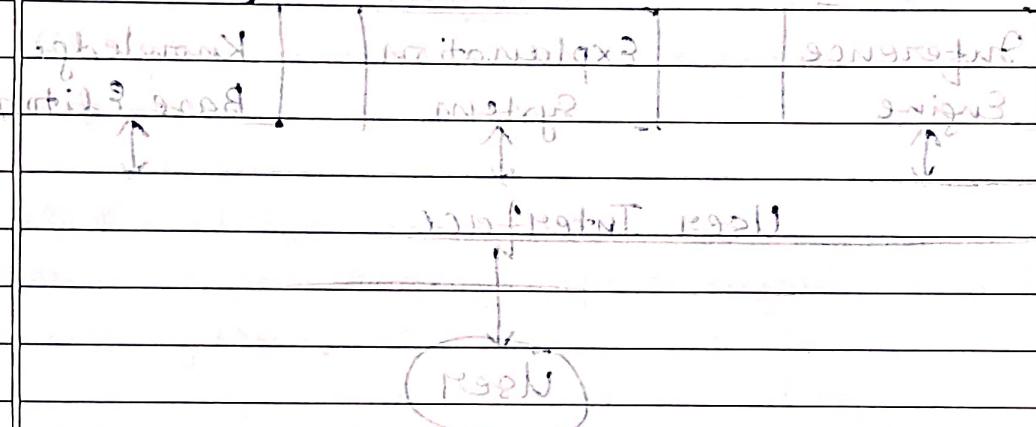
Key components in NLP

- (1) Language Understanding
- (2) Machine Translation
- (3) Chatbots & conversational AI
- (4) Summarization
- (5) Generation

## (6) Sentiment Analysis & Opinion mining (F)

## (7) Ethical & Bias Considerations

NLP continues to play a pivotal role in various applications, including customer service, healthcare, finance, education & more.



(8) Significant development in NLP includes:

Introducing multilingual models to support multiple languages & dialects, incorporating more robust handling of language & grammar, integrating multimodal inputs (e.g., images, audio), and developing more ethical AI systems.

Significant developments in NLP include:

Introducing multilingual models to support multiple languages & dialects, incorporating more robust handling of language & grammar, integrating multimodal inputs (e.g., images, audio), and developing more ethical AI systems.

Significant developments in NLP include:

Introducing multilingual models to support multiple languages & dialects, incorporating more robust handling of language & grammar, integrating multimodal inputs (e.g., images, audio), and developing more ethical AI systems.