

# Lab Assignment 7

**Subject:** Artificial Intelligence

**Guided by:** Dr. Anuradha Yenikar

**Student name:** Neha Sawant

---

**Experiment Name:** Implement Forward Chaining Algorithm.

## Objective:

The objective of this lab assignment is to implement the forward chaining algorithm, a fundamental technique in artificial intelligence used for reasoning in rule-based systems. The goal is to infer new facts based on a set of initial facts and defined rules.

## Problem Statement:

In this exercise, you will simulate a simple expert system that uses forward chaining to infer new information from given facts and rules. The system will begin with a set of known facts and apply rules iteratively to derive new facts until no further conclusions can be drawn.

## Requirements:

- Programming Language: Python
- Environment: Any Python IDE (e.g., PyCharm, Jupyter Notebook)
- Python Version: 3.6 or higher

## Code Explanation:

### 1] Code Overview

The provided code defines a simple forward chaining algorithm that uses initial facts and a set of rules to infer new facts.

### 2] Components of the Code

- Facts: A list of known facts. In this case, the facts are ``["A", "B"]``.

- Rules: A list of tuples, where each tuple contains a list of conditions and a conclusion.
- For example, `(["A", "B"], "C")` means that if both A and B are true, then C can be inferred.

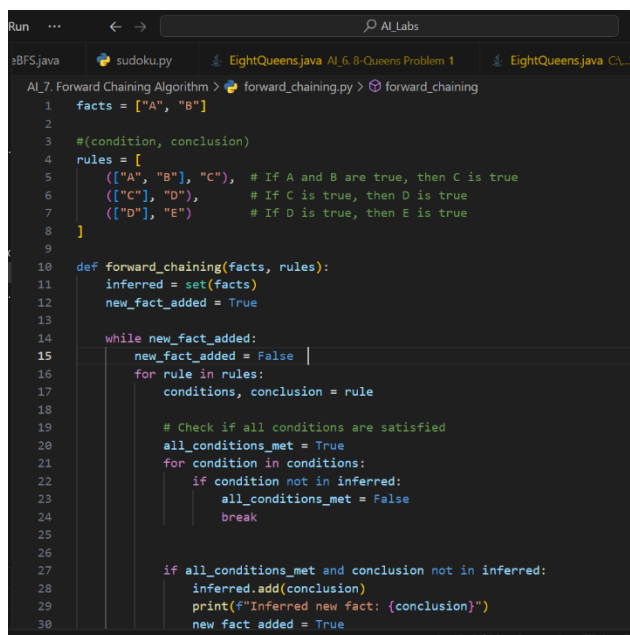
- forward\_chaining(facts, rules):

This function implements the forward chaining algorithm:

- It starts with the initial facts and tries to infer new facts using the provided rules.
- The algorithm continues until no new facts can be inferred.

### 3] Code Implementation

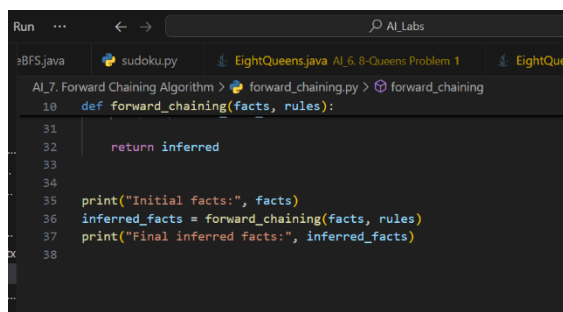
Here's the complete implementation of the forward chaining algorithm:



```

1  facts = ["A", "B"]
2
3  #(condition, conclusion)
4  rules = [
5      (["A", "B"], "C"), # If A and B are true, then C is true
6      (["C"], "D"),      # If C is true, then D is true
7      (["D"], "E")       # If D is true, then E is true
8  ]
9
10 def forward_chaining(facts, rules):
11     inferred = set(facts)
12     new_fact_added = True
13
14     while new_fact_added:
15         new_fact_added = False
16         for rule in rules:
17             conditions, conclusion = rule
18
19             # Check if all conditions are satisfied
20             all_conditions_met = True
21             for condition in conditions:
22                 if condition not in inferred:
23                     all_conditions_met = False
24                     break
25
26             if all_conditions_met and conclusion not in inferred:
27                 inferred.add(conclusion)
28                 print(f"Inferred new fact: {conclusion}")
29                 new_fact_added = True
30

```



```

10 def forward_chaining(facts, rules):
31
32     return inferred
33
34
35 print("Initial facts:", facts)
36 inferred_facts = forward_chaining(facts, rules)
37 print("Final inferred facts:", inferred_facts)
38

```

### 4] Output

The program will output the initial facts, any newly inferred facts, and the final set of inferred facts. An example output might look like this:

```
PS C:\Users\nehas\Downloads\AI_Labs> python -u "c:\Users\nehas\Downloads\AI_Labs\forward_chaining.py"
Initial facts: ['A', 'B']
Inferred new fact: C
Inferred new fact: D
Inferred new fact: E
Final inferred facts: {'E', 'B', 'A', 'C', 'D'}
PS C:\Users\nehas\Downloads\AI_Labs>
```

### Working of the Code:

1. Initialization: The initial facts are defined, and a set of rules is established.
2. Inference Process:
  - The algorithm uses a while loop to continue inferring new facts until no new facts can be added.
  - For each rule, it checks if all conditions are satisfied. If they are and the conclusion is not already inferred, it adds the conclusion to the set of inferred facts.
  - The process repeats until no new conclusions can be drawn.
3. Result Display: Finally, the inferred facts are printed.

### Conclusion:

In this lab, I have successfully implemented the forward chaining algorithm to infer new information in a rule-based system. This exercise has enhanced my understanding of rule-based reasoning and how to implement logical inference using programming.