

Forward Chaining

Aim:

To implement Forward Chaining

Code:

```
def forward_chaining(rules, facts):
    inferred_facts = set(facts) # Start with initial facts
    newly_inferred = True
    while newly_inferred:
        newly_inferred = False
        for antecedent, consequent in rules:
            if all(fact in inferred_facts for fact in antecedent):
                for fact in consequent: # Add each consequent fact
                    if fact not in inferred_facts:
                        inferred_facts.add(fact)
                        newly_inferred = True

    return inferred_facts

rules = [
    (["A", "B"], ["C"]),
    (["C", "D"], ["E"]),
    (["A"], ["B"]),
    (["E"], ["F"]),
]

facts = {"A", "D"}
inferred_facts = forward_chaining(rules, facts)
print("Initial facts:", facts)
print("Inferred facts:", inferred_facts)
```

Output:

Initial facts: {'D', 'A'}

Inferred facts: {'F', 'A', 'D', 'B', 'C', 'E'}

Result:

Forward Chaining implemented successfully.