Ex No: 2c

IMPLEMENTATION OF BACKWARD CHAINING

Aim*:*

To implement backward chaining.

Scenario:

A medical expert system is designed to **diagnose diseases** based on patient symptoms. The system uses **backward chaining** to infer whether a patient has a specific disease by checking rules and known facts.

Procedure:

1. **Define the knowledge base** with **rules** (causal relationships).

   - `"flu": [["cough", "fever"]]` → Flu occurs if both **cough** and **fever** exist.
   - `"fever": [["sore_throat"]]` → Fever occurs if **sore throat** exists.

2. **Define known facts**: `{sore_throat, cough}`.

3. **Define the backward chaining function**:

   - **Check if the goal is in known facts.** If so, return `True`.
   - **Check if rules exist for the goal in the knowledge base.**
   - **For each rule**, verify **all conditions recursively** using backward chaining.
   - If all conditions **can be proven**, return `True`.
   - Otherwise, return `False`.

4. **Query whether the patient has flu (`flu`).**

5. **Execution:**

   - `flu` requires `cough` and `fever`.
   - `cough` is a **fact → True**
   - `fever` needs `sore_throat`.
   - `sore_throat` is a **fact → True**
   - Since both `cough` and `fever` are proven **flu is diagnosed**.

Program:

```
# Knowledge Base (Rules in IF-THEN format)
knowledge_base = {
    "flu": [["cough", "fever"]],
    "fever": [["sore_throat"]],
}
# Known facts
facts = {"sore_throat", "cough"}
# Backward chaining function
def backward_chaining(goal):
    if goal in facts:  # If the goal is a known fact, return True
        return True
    if goal in knowledge_base:  # If the goal has rules in KB
        for conditions in knowledge_base[goal]:  # Check each rule
            if all(backward_chaining(cond) for cond in conditions):  # Recursively verify
                return True
    return False  # If no rule or fact supports the goal, return False
# Query: Does the patient have flu?
query = "flu"
if backward_chaining(query):
    print(f"The patient is diagnosed with {query}.")
else:
    print(f"The patient does NOT have {query}.")
```

Output:

```
The patient is diagnosed with flu.
```