# LAB - 07 :-

DATE: 28 PAGE: 12/11/24

* Solve :

1. Alice is the mother of bob
2. Bob is the father of charlie
3. A father is a parent
4. A mother is a parent
5. All parents have children
6. If someone is a parent, their children are siblings.
7. Alice is married to david.

# Hypothesis :

Charlie is sibling of bob

→ Step by step

* Identify Relationships :

From (1) Alice is bob's mother

From (2) Bob is charlie's father

From (3) & (4) both mothers & fathers are parents

From (5) as parents, both Alice & bob have children

& from (6) if someone is a parent, their children are considered siblings.

Since bob is charlie's father, Charlie would be considered bob's child in a sibling.

conclusion :-

(c) Bob as a parent, implies that his children Charlie would be siblings if there were other children.

DATE: PAGE:

A : Alice is the mother of bob
C : charlie is a child of bob
S : children of parent are siblings
H : Hypothesis - charlie is a sibling of Bob

| A | C | S | H | conclusion |
|---|---|---|---|---|
| T | T | T | T | True |
| T | T | F | F | False |
| T | F | T | F | False |
| F | T | T | F | False |
| F | F | F | F | False |

**LAB 07 :**

**check entailment :**

CODE:

```
# Function to check entailment based on user input
def check_entailment():
    print("Welcome to the Entailment Checker!")

    # Step 1: Gather user input for facts (Premises)
    alice_is_mother_of_bob = input("Enter the fact: Alice is the mother
of Bob. (e.g., 'Alice is the mother of Bob')\n")
    bob_is_father_of_charlie = input("Enter the fact: Bob is the father
of Charlie. (e.g., 'Bob is the father of Charlie')\n")
    father_is_parent = input("Enter the fact: A father is a parent.
(e.g., 'A father is a parent')\n")
    mother_is_parent = input("Enter the fact: A mother is a parent.
(e.g., 'A mother is a parent')\n")
    all_parents_have_children = input("Enter the fact: All parents have
children. (e.g., 'All parents have children')\n")
    parents_children_are_siblings = input("Enter the fact: Parents'
children are siblings. (e.g., 'Parents' children are siblings')\n")
    alice_is_married_to_david = input("Enter the fact: Alice is married
to David. (e.g., 'Alice is married to David')\n")

    # Step 2: Entailment reasoning process
    if ('Alice is the mother of Bob' in alice_is_mother_of_bob and
        'Bob is the father of Charlie' in bob_is_father_of_charlie and
        'A father is a parent' in father_is_parent and
        'A mother is a parent' in mother_is_parent and
        'All parents have children' in all_parents_have_children and
        "Parents' children are siblings" in
parents_children_are_siblings and
        'Alice is married to David' in alice_is_married_to_david):

        # Conclusion: Check if Charlie is a sibling of Bob
        print("\nSince Alice is Bob's mother and Bob is Charlie's
father, Charlie and Bob are siblings.")
        print("Conclusion: Charlie is a sibling of Bob. The hypothesis
is entailed by the knowledge base.")
    else:
        print("\nThe information provided does not fully support the
conclusion.")

# Run the function
check_entailment()
```

output:

```
Welcome to the Entailment Checker!
Enter the fact: Alice is the mother of Bob. (e.g., 'Alice is the mother of Bob')
 Alice is the mother of Bob
Enter the fact: Bob is the father of Charlie. (e.g., 'Bob is the father of Charlie')
Bob is the father of Charlie
Enter the fact: A father is a parent. (e.g., 'A father is a parent')
A father is a parent
Enter the fact: A mother is a parent. (e.g., 'A mother is a parent')
 A mother is a parent
Enter the fact: All parents have children. (e.g., 'All parents have children')
All parents have children
Enter the fact: Parents' children are siblings. (e.g., 'Parents' children are siblings')
Parents' children are siblings
Enter the fact: Alice is married to David. (e.g., 'Alice is married to David')
Alice is married to David

Since Alice is Bob's mother and Bob is Charlie's father, Charlie and Bob are siblings.
Conclusion: Charlie is a sibling of Bob. The hypothesis is entailed by the knowledge base.
```