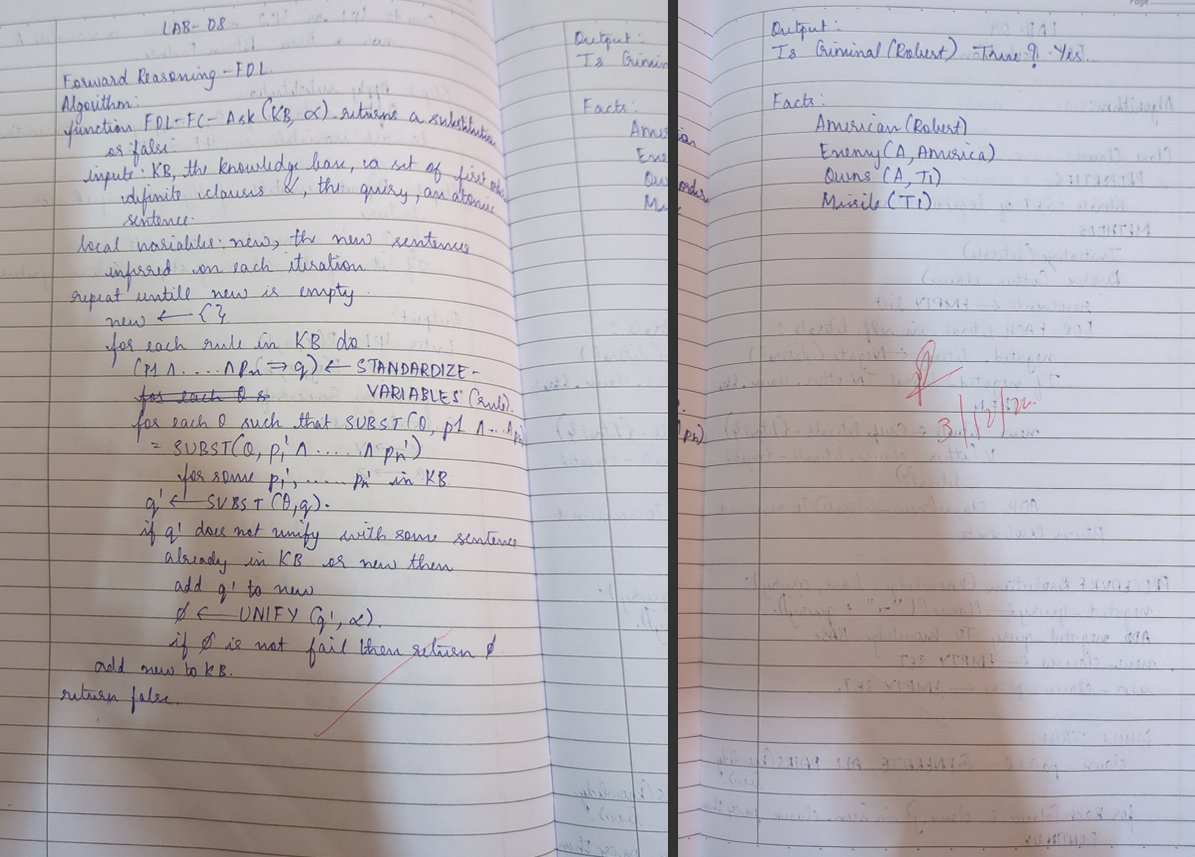
**First Order Logic – Forward Reasoning**

**Algorithm:**



**Code:**

class KnowledgeBase:

    def \_\_init\_\_(self):

        self.facts = set()

        self.rules = []

    def add\_fact(self, fact):

        self.facts.add(fact)

    def add\_rule(self, rule):

        self.rules.append(rule)

    def forward\_reason(self):

        new\_facts = set()

        while True:

            for rule in self.rules:

                inferred = rule(self.facts)

                # Add only new facts that aren't already in the knowledge base

                new\_facts.update(inferred - self.facts)

            if not new\_facts:

                break

            self.facts.update(new\_facts)

            new\_facts.clear()

    def query(self, fact):

        return fact in self.facts

def rule\_american\_criminal(facts):

    inferred = set()

    for fact in facts:

        if fact.startswith("Sells("):  # Find sells facts

            parts = fact[6:-1].split(", ")  # Extract parts: Sells(person, weapon, country)

            person, weapon, country = parts[0], parts[1], parts[2]

            if f"American({person})" in facts and f"Weapon({weapon})" in facts and f"Hostile({country})" in facts:

                inferred.add(f"Criminal({person})")

    return inferred

def rule\_hostile\_enemy(facts):

    inferred = set()

    for fact in facts:

        if fact.startswith("Enemy("):

            parts = fact[6:-1].split(", ")  # Extract parts: Enemy(country, America)

            country = parts[0]

            inferred.add(f"Hostile({country})")

    return inferred

def rule\_weapons\_from\_missiles(facts):

    inferred = set()

    for fact in facts:

        if fact.startswith("Missile("):

            missile = fact[8:-1]  # Extract missile name

            inferred.add(f"Weapon({missile})")

    return inferred

def rule\_sells\_missiles(facts):

    inferred = set()

    for fact in facts:

        if fact.startswith("Owns("):

            parts = fact[5:-1].split(", ")  # Extract parts: Owns(country, item)

            country, item = parts[0], parts[1]

            if f"Missile({item})" in facts:

                inferred.add(f"Sells(Robert, {item}, {country})")

    return inferred

kb = KnowledgeBase()

kb.add\_fact("American(Robert)")

kb.add\_fact("Enemy(A, America)")

kb.add\_fact("Owns(A, T1)")

kb.add\_fact("Missile(T1)")

kb.add\_rule(rule\_american\_criminal)

kb.add\_rule(rule\_hostile\_enemy)

kb.add\_rule(rule\_weapons\_from\_missiles)

kb.add\_rule(rule\_sells\_missiles)

kb.forward\_reason()

query = "Criminal(Robert)"

print(f"Is '{query}' true? \n{'Yes' if kb.query(query) else 'No'}")

**Output:**

