IMPLEMENTION OF FORWARD CHAINING

```
facts = {
    'a': True,
    'b': True,
    'c': False
}
rules = [
    ('d', ['a', 'b']),  # d can be concluded if a and b are true
    ('e', ['b', 'c']),  # e can be concluded if b and c are true
    ('f', ['d', 'e'])   # f can be concluded if d and e are true
]
def forward_chaining(facts, rules, goal):
    inferred = set([fact for fact, value in facts.items() if value])  # Only include true facts
    new_inferred = inferred.copy()

    while new_inferred:
        current_inferred = set()
        for head, body in rules:
            if head not in inferred and all(fact in inferred for fact in body):
                current_inferred.add(head)
        if current_inferred:
            inferred.update(current_inferred)
            new_inferred = current_inferred
        else:
            break
    return goal in inferred
goal = 'f'

if forward_chaining(facts, rules, goal):
```

```
    print(f"The goal '{goal}' can be achieved.")
else:
    print(f"The goal '{goal}' cannot be achieved.")
```

OUTPUT:

The goal ‘f’ can be achieved.

The goal ‘e’ cannot be achieved.

The goal ‘d’ can be achieved.