IMPLEMENTATION OF BACKWARD CHAINING

SOURCE CODE:

```
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 backward_chaining(goal, facts, rules):
    # Check if the goal is already a fact
    if goal in facts:
        return facts[goal]

    for rule in rules:
        head, body = rule
        if head == goal:
            # Recursively check if all conditions in the body are true
            if all(backward_chaining(cond, facts, rules) for cond in body):
                return True

    return False

goal = 'f'
```

```
if backward_chaining(goal, facts, rules):
    print(f"The goal '{goal}' can be achieved.")
else:
    print(f"The goal '{goal}' cannot be achieved.")
```

OUTPUT:

The goal 'f' cannot be achieved.

Expected Output for the Goal ‘f’:

The goal ‘f’ cannot be achieved.

Expected Output for the Goal ‘d’:

The goal ‘d’ can be achieved.