Q1.write about FIND S theorem with implementation code in python. ¶

The FIND-S algorithm is a machine learning algorithm used for learning a hypothesis from a given set of training examples. It is a simple and efficient algorithm that is widely used for binary classification tasks. The algorithm starts with the most specific hypothesis and then generalizes it until it covers all positive training examples. The most specific hypothesis is usually represented by a conjunction of literals, where each literal represents a feature-value pair.

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
a = [
    ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same'],
    ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same'],
    ['Rainy', 'Cold', 'High', 'Strong', 'Warm', 'Change'],
    ['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change']
]
```

In [3]:
```
a
```

Out[3]:
```
[['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same'],
 ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same'],
 ['Rainy', 'Cold', 'High', 'Strong', 'Warm', 'Change'],
 ['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change']]
```

In [4]:
```
t = ['Yes', 'Yes', 'No', 'Yes']
```

In [5]:
```
a,t
```

Out[5]:
```
([['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same'],
  ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same'],
  ['Rainy', 'Cold', 'High', 'Strong', 'Warm', 'Change'],
  ['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change']],
 ['Yes', 'Yes', 'No', 'Yes'])
```

In [6]:
```
def fun(c, t):
    specific_hypothesis = None
    for i, val in enumerate(t):
        if val == "Yes":
            specific_hypothesis = c[i].copy()
            break

    if specific_hypothesis is None:
        print("No positive examples found in target dataset.")
        return None

    for i, val in enumerate(c):
        if t[i] == "Yes":
            for x in range(len(specific_hypothesis)):
                if val[x] != specific_hypothesis[x]:
                    specific_hypothesis[x] = '?'
                else:
                    pass

    return specific_hypothesis
```

In [7]:
```
print("The final hypothesis is:", fun(a, t))
```

```
The final hypothesis is: ['Sunny', 'Warm', '?', 'Strong', '?', '?']
```

In [ ]:

In [ ]: