

```

import csv
from tabulate import tabulate

def load_data(filename):
    with open(filename, 'r') as file:
        reader = csv.reader(file)
        all_rows = list(reader)
        header = all_rows[0]
        data = all_rows[1:]
    return header, data

def more_general(h1, h2):
    """Returns True if h1 is more general than h2."""
    for x, y in zip(h1, h2):
        if x != '?' and (x != y and y != '?'):
            return False
    return True

def candidate_elimination(data):
    n_attr = len(data[0]) - 1
    S = data[0][:n_attr]
    G = [['?'] * n_attr]

    for example in data:
        instance, label = example[:n_attr], example[n_attr]
        if label == 'Yes':
            # Remove inconsistent hypotheses from G
            G = [g for g in G if more_general(g, instance)]

            for i in range(len(S)):
                if S[i] != instance[i]:
                    S[i] = '?'
        else:
            G_temp = []
            for g in G:
                for i in range(len(g)):
                    if g[i] == '?':
                        if S[i] != '?':
                            g_new = g.copy()
                            g_new[i] = S[i]
                            if g_new not in G_temp:
                                G_temp.append(g_new)

            G = G_temp

    return S, G

# Load training data from CSV
header, training_data = load_data("training_data.csv")

# Print table of training data
print("\nTraining Data:\n")
print(tabulate(training_data, headers=header, tablefmt="grid"))

# Run Candidate Elimination

```

```
S_final, G_final = candidate_elimination(training_data)

# Output
print("\nFinal Specific Hypothesis (S):", S_final)
print("\nFinal General Hypotheses (G):")
for g in G_final:
    print(g)
```