```
import math
from collections import Counter

data = [
 ['Sunny','Hot','High','Weak','No'],
 ['Sunny','Hot','High','Strong','No'],
 ['Overcast','Hot','High','Weak','Yes'],
 ['Rain','Mild','High','Weak','Yes'],
 ['Rain','Cool','Normal','Weak','Yes'],
 ['Rain','Cool','Normal','Strong','No'],
 ['Overcast','Cool','Normal','Strong','Yes'],
 ['Sunny','Mild','High','Weak','No'],
 ['Sunny','Cool','Normal','Weak','Yes'],
 ['Rain','Mild','Normal','Weak','Yes'],
 ['Sunny','Mild','Normal','Strong','Yes'],
 ['Overcast','Mild','High','Strong','Yes'],
 ['Overcast','Hot','Normal','Weak','Yes'],
 ['Rain','Mild','High','Strong','No']
]

attrs = ['Outlook','Temp','Humidity','Wind']

entropy = lambda d: -sum((c/len(d))*math.log2(c/len(d))
                        for c in Counter(r[-1] for r in d).values())

gain = lambda d,i: entropy(d) - sum(
    (len(s)/len(d))*entropy(s)
    for v in set(r[i] for r in d)
    for s in [[r for r in d if r[i]==v]]
)
```

```
def id3(d, a):
    if len(set(r[-1] for r in d)) == 1:
        return d[0][-1]
    if not a:
        return Counter(r[-1] for r in d).most_common(1)[0][0]

    i = max(range(len(a)), key=lambda x: gain(d, x))
    return {a[i]: {v: id3([r[:i]+r[i+1:] for r in d if r[i]==v],
                          a[:i]+a[i+1:])
                   for v in set(r[i] for r in d)}}

def classify(t, a, x):
    return t if not isinstance(t, dict) else classify(
        t[next(iter(t))][x[a.index(next(iter(t)))]], a, x)

tree = id3(data, attrs)
print("Decision Tree:", tree)

sample = ['Sunny','Cool','High','Strong']
print("Prediction:", classify(tree, attrs, sample))
```

OUTPUT: