**4.** **Write a program to demonstrate the working of the decision tree based ID3**

**algorithm. Use an appropriate data set for building the decision tree and apply**

**this knowledge to classify a new sample.**

THEORY: The ID3 algorithm begins with the original set as the root node. On each iteration of the algorithm, it iterates through every unused attribute of the set and calculates the entropy or the information gain of that attribute. It then selects the attribute which has the smallest entropy (or largest information gain) value. The set is then split or partitioned by the selected attribute to produce subsets of the data. (For example, a node can be split into child nodes based upon the subsets of the population whose ages are less than 50, between 50 and 100, and greater than 100.) The algorithm continues to recurse on each subset, considering only attributes never selected before.

* Calculate the entropy of every attribute {\displaystyle a}a of the data set {\displaystyle S}S.
* Partition ("split") the set {\displaystyle S}S into subsets using the attribute for which the resulting entropy after splitting is minimized; or, equivalently, information gain is maximum.
* Make a decision tree node containing that attribute.
* Recurse on subsets using the remaining attributes.

The ID3 algorithm is used by training on a data set {\displaystyle S}S to produce a decision tree which is stored in memory. At runtime, this decision tree is used to classify new test cases (feature vectors) by traversing the decision tree using the features of the datum to arrive at a leaf node.

**PROCEDURE/PROGRAM:**

import pandas as pd

df = pd.read\_csv('PlayTennis.csv')

print("\n Input Data Set is:\n", df)

Input Data Set is:

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

2 overcast hot high weak yes

3 rain mild high weak yes

4 rain cool normal weak yes

5 rain cool normal strong no

6 overcast cool normal strong yes

7 sunny mild high weak no

8 sunny cool normal weak yes

9 rain mild normal weak yes

10 sunny mild normal strong yes

11 overcast mild high strong yes

12 overcast hot normal weak yes

13 rain mild high strong no

t=df.keys()[-1]

print('Target Attribute is: ', t)

attribute\_names = list(df.keys())

attribute\_names.remove(t)

print('Predicting Attributes: ', attribute\_names)

Target Attribute is: playtennis

Predicting Attributes: ['outlook', 'tempearture', 'humidity', 'wind']

import math

**def** entropy(probs):

    return sum( [-prob\*math.log(prob, 2) for prob in probs])

**def** entropy\_of\_list(ls,value):

    from collections import Counter

    cnt = Counter(x for x in ls)

    print('Target attribute class count(Yes/No)=',dict(cnt))

    total\_instances = len(ls)

    print("Total no of instances/records associated with {0} is: {1}".format(value,total\_instances ))

    probs = [x / total\_instances for x in cnt.values()]

    print("Probability of Class {0} is: {1**:.4f**}".format(min(cnt),min(probs)))

    print("Probability of Class {0} is: {1**:.4f**}".format(max(cnt),max(probs)))

    return entropy(probs)

**def** information\_gain(df, split\_attribute, target\_attribute,battr):

    print("\n\n-----Information Gain Calculation of ",split\_attribute, " --------")

    df\_split = df.groupby(split\_attribute)

    glist=[]

    for gname,group in df\_split:

        print('Grouped Attribute Values \n',group)

        glist.append(gname)

    glist.reverse()

    nobs = len(df.index) \* 1.0

    df\_agg1=df\_split.agg({target\_attribute:**lambda** x:entropy\_of\_list(x, glist.pop())})

    df\_agg2=df\_split.agg({target\_attribute :**lambda** x:len(x)/nobs})

    df\_agg1.columns=['Entropy']

    df\_agg2.columns=['Proportion']

    new\_entropy = sum( df\_agg1['Entropy'] \* df\_agg2['Proportion'])

    if battr !='S':

        old\_entropy = entropy\_of\_list(df[target\_attribute],'S-'+df.iloc[0][df.columns.get\_loc(battr)])

    else:

        old\_entropy = entropy\_of\_list(df[target\_attribute],battr)

    return old\_entropy - new\_entropy

**def** id3(df,target\_attribute,attribute\_names,default\_class=None,default\_attr='S'):

    from collections import Counter

    cnt = Counter(x for x in df[target\_attribute])

    if len(cnt)==1:

        return next(iter(cnt))

    elif df.empty or (not attribute\_names):

        return default\_class

    else:

        default\_class=max(cnt.keys())

        gainz=[]

        for attr in attribute\_names:

            ig=information\_gain(df,attr,target\_attribute,default\_attr)

            gainz.append(ig)

            print('Information gain of',attr,'is:',ig)

        index\_of\_max=gainz.index(max(gainz))

        best\_attr=attribute\_names[index\_of\_max]

        print("\nAttribute with the maximum gain is:",best\_attr)

        tree={best\_attr:{}}

        remaining\_attribute\_names=[i for i in attribute\_names if i!=best\_attr]

        for attr\_val,data\_subset in df.groupby (best\_attr):

            subtree=id3(data\_subset,target\_attribute,remaining\_attribute\_names,default\_class,best\_attr)

            tree[best\_attr][attr\_val]=subtree

        return tree

from pprint import pprint

tree = id3(df,t,attribute\_names)

print("\nthe Resultant Decision Tree is:")

pprint(tree)

-----Information Gain Calculation of outlook --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

2 overcast hot high weak yes

6 overcast cool normal strong yes

11 overcast mild high strong yes

12 overcast hot normal weak yes

Grouped Attribute Values

outlook tempearture humidity wind playtennis

3 rain mild high weak yes

4 rain cool normal weak yes

5 rain cool normal strong no

9 rain mild normal weak yes

13 rain mild high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

7 sunny mild high weak no

8 sunny cool normal weak yes

10 sunny mild normal strong yes

Target attribute class count(Yes/No)= {'yes': 4}

Total no of instances/records associated with overcast is: 4

Probability of Class yes is: 1.0000

Probability of Class yes is: 1.0000

Target attribute class count(Yes/No)= {'yes': 3, 'no': 2}

Total no of instances/records associated with rain is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Target attribute class count(Yes/No)= {'no': 3, 'yes': 2}

Total no of instances/records associated with sunny is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Target attribute class count(Yes/No)= {'no': 5, 'yes': 9}

Total no of instances/records associated with S is: 14

Probability of Class no is: 0.3571

Probability of Class yes is: 0.6429

Information gain of outlook is: 0.2467498197744391

-----Information Gain Calculation of tempearture --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

4 rain cool normal weak yes

5 rain cool normal strong no

6 overcast cool normal strong yes

8 sunny cool normal weak yes

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

2 overcast hot high weak yes

12 overcast hot normal weak yes

Grouped Attribute Values

outlook tempearture humidity wind playtennis

3 rain mild high weak yes

7 sunny mild high weak no

9 rain mild normal weak yes

10 sunny mild normal strong yes

11 overcast mild high strong yes

13 rain mild high strong no

Target attribute class count(Yes/No)= {'yes': 3, 'no': 1}

Total no of instances/records associated with cool is: 4

Probability of Class no is: 0.2500

Probability of Class yes is: 0.7500

Target attribute class count(Yes/No)= {'no': 2, 'yes': 2}

Total no of instances/records associated with hot is: 4

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'yes': 4, 'no': 2}

Total no of instances/records associated with mild is: 6

Probability of Class no is: 0.3333

Probability of Class yes is: 0.6667

Target attribute class count(Yes/No)= {'no': 5, 'yes': 9}

Total no of instances/records associated with S is: 14

Probability of Class no is: 0.3571

Probability of Class yes is: 0.6429

Information gain of tempearture is: 0.029222565658954647

-----Information Gain Calculation of humidity --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

2 overcast hot high weak yes

3 rain mild high weak yes

7 sunny mild high weak no

11 overcast mild high strong yes

13 rain mild high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

4 rain cool normal weak yes

5 rain cool normal strong no

6 overcast cool normal strong yes

8 sunny cool normal weak yes

9 rain mild normal weak yes

10 sunny mild normal strong yes

12 overcast hot normal weak yes

Target attribute class count(Yes/No)= {'no': 4, 'yes': 3}

Total no of instances/records associated with high is: 7

Probability of Class no is: 0.4286

Probability of Class yes is: 0.5714

Target attribute class count(Yes/No)= {'yes': 6, 'no': 1}

Total no of instances/records associated with normal is: 7

Probability of Class no is: 0.1429

Probability of Class yes is: 0.8571

Target attribute class count(Yes/No)= {'no': 5, 'yes': 9}

Total no of instances/records associated with S is: 14

Probability of Class no is: 0.3571

Probability of Class yes is: 0.6429

Information gain of humidity is: 0.15183550136234136

-----Information Gain Calculation of wind --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

1 sunny hot high strong no

5 rain cool normal strong no

6 overcast cool normal strong yes

10 sunny mild normal strong yes

11 overcast mild high strong yes

13 rain mild high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

2 overcast hot high weak yes

3 rain mild high weak yes

4 rain cool normal weak yes

7 sunny mild high weak no

8 sunny cool normal weak yes

9 rain mild normal weak yes

12 overcast hot normal weak yes

Target attribute class count(Yes/No)= {'no': 3, 'yes': 3}

Total no of instances/records associated with strong is: 6

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'no': 2, 'yes': 6}

Total no of instances/records associated with weak is: 8

Probability of Class no is: 0.2500

Probability of Class yes is: 0.7500

Target attribute class count(Yes/No)= {'no': 5, 'yes': 9}

Total no of instances/records associated with S is: 14

Probability of Class no is: 0.3571

Probability of Class yes is: 0.6429

Information gain of wind is: 0.04812703040826927

Attribute with the maximum gain is: outlook

-----Information Gain Calculation of tempearture --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

4 rain cool normal weak yes

5 rain cool normal strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

3 rain mild high weak yes

9 rain mild normal weak yes

13 rain mild high strong no

Target attribute class count(Yes/No)= {'yes': 1, 'no': 1}

Total no of instances/records associated with cool is: 2

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'yes': 2, 'no': 1}

Total no of instances/records associated with mild is: 3

Probability of Class no is: 0.3333

Probability of Class yes is: 0.6667

Target attribute class count(Yes/No)= {'yes': 3, 'no': 2}

Total no of instances/records associated with S-rain is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of tempearture is: 0.01997309402197489

-----Information Gain Calculation of humidity --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

3 rain mild high weak yes

13 rain mild high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

4 rain cool normal weak yes

5 rain cool normal strong no

9 rain mild normal weak yes

Target attribute class count(Yes/No)= {'yes': 1, 'no': 1}

Total no of instances/records associated with high is: 2

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'yes': 2, 'no': 1}

Total no of instances/records associated with normal is: 3

Probability of Class no is: 0.3333

Probability of Class yes is: 0.6667

Target attribute class count(Yes/No)= {'yes': 3, 'no': 2}

Total no of instances/records associated with S-rain is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of humidity is: 0.01997309402197489

-----Information Gain Calculation of wind --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

5 rain cool normal strong no

13 rain mild high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

3 rain mild high weak yes

4 rain cool normal weak yes

9 rain mild normal weak yes

Target attribute class count(Yes/No)= {'no': 2}

Total no of instances/records associated with strong is: 2

Probability of Class no is: 1.0000

Probability of Class no is: 1.0000

Target attribute class count(Yes/No)= {'yes': 3}

Total no of instances/records associated with weak is: 3

Probability of Class yes is: 1.0000

Probability of Class yes is: 1.0000

Target attribute class count(Yes/No)= {'yes': 3, 'no': 2}

Total no of instances/records associated with S-rain is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of wind is: 0.9709505944546686

Attribute with the maximum gain is: wind

-----Information Gain Calculation of tempearture --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

8 sunny cool normal weak yes

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

7 sunny mild high weak no

10 sunny mild normal strong yes

Target attribute class count(Yes/No)= {'yes': 1}

Total no of instances/records associated with cool is: 1

Probability of Class yes is: 1.0000

Probability of Class yes is: 1.0000

Target attribute class count(Yes/No)= {'no': 2}

Total no of instances/records associated with hot is: 2

Probability of Class no is: 1.0000

Probability of Class no is: 1.0000

Target attribute class count(Yes/No)= {'no': 1, 'yes': 1}

Total no of instances/records associated with mild is: 2

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'no': 3, 'yes': 2}

Total no of instances/records associated with S-sunny is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of tempearture is: 0.5709505944546686

-----Information Gain Calculation of humidity --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

1 sunny hot high strong no

7 sunny mild high weak no

Grouped Attribute Values

outlook tempearture humidity wind playtennis

8 sunny cool normal weak yes

10 sunny mild normal strong yes

Target attribute class count(Yes/No)= {'no': 3}

Total no of instances/records associated with high is: 3

Probability of Class no is: 1.0000

Probability of Class no is: 1.0000

Target attribute class count(Yes/No)= {'yes': 2}

Total no of instances/records associated with normal is: 2

Probability of Class yes is: 1.0000

Probability of Class yes is: 1.0000

Target attribute class count(Yes/No)= {'no': 3, 'yes': 2}

Total no of instances/records associated with S-sunny is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of humidity is: 0.9709505944546686

-----Information Gain Calculation of wind --------

Grouped Attribute Values

outlook tempearture humidity wind playtennis

1 sunny hot high strong no

10 sunny mild normal strong yes

Grouped Attribute Values

outlook tempearture humidity wind playtennis

0 sunny hot high weak no

7 sunny mild high weak no

8 sunny cool normal weak yes

Target attribute class count(Yes/No)= {'no': 1, 'yes': 1}

Total no of instances/records associated with strong is: 2

Probability of Class no is: 0.5000

Probability of Class yes is: 0.5000

Target attribute class count(Yes/No)= {'no': 2, 'yes': 1}

Total no of instances/records associated with weak is: 3

Probability of Class no is: 0.3333

Probability of Class yes is: 0.6667

Target attribute class count(Yes/No)= {'no': 3, 'yes': 2}

Total no of instances/records associated with S-sunny is: 5

Probability of Class no is: 0.4000

Probability of Class yes is: 0.6000

Information gain of wind is: 0.01997309402197489

Attribute with the maximum gain is: humidity

the Resultant Decision Tree is:

{'outlook': {'overcast': 'yes',

'rain': {'wind': {'strong': 'no', 'weak': 'yes'}},

'sunny': {'humidity': {'high': 'no', 'normal': 'yes'}}}}

**def** classify(instance,tree,default=None):

    attribute=next(iter(tree))

    if instance[attribute] in tree[attribute].keys():

        result = tree[attribute][instance[attribute]]

        if isinstance (result,dict):

            return classify(instance,result)

        else:

            return result

    else:

        return default

df\_new = pd.read\_csv('PlayTennisTest.csv')

df\_new['predicted'] = df\_new.apply(classify,axis=1,args=(tree,'?'))

print(df\_new)

outlook temperature humidity wind playtennis predicted

0 sunny hot normal weak ? yes

1 rain mild high strong ? no