Name : Devkumar Biswas

Class: BE(AI&DS)

Div: B

Subject : IR(CL-II Lab)

Roll no. : BEAD21267

**Assignment No. - 3**

**Problem Statement** : Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using the standard Heart Disease Data Set (You can use Java/Python ML library classes/API.

Code:-

import pandas as pd

import numpy as np

# Sample Heart Disease Data Set

data = {

'age': [63, 67, 67, 37, 41],

'sex': [1, 1, 1, 1, 0],

'cp': [3, 2, 2, 3, 1],

'trestbps': [145, 160, 120, 130, 130],

'chol': [233, 286, 229, 250, 204],

'fbs': [1, 0, 0, 0, 0],

'restecg': [0, 2, 0, 0, 2],

'thalach': [150, 108, 129, 187, 172],

'exang': [0, 1, 1, 0, 0],

'oldpeak': [2.3, 1.5, 2.6, 3.5, 1.4],

'slope': [0, 1, 0, 0, 1],

'ca': [0, 3, 2, 0, 0],

'thal': [1, 2, 2, 3, 2],

'heartdisease': [1, 1, 1, 0, 0]

}

# Convert to DataFrame

heart\_disease = pd.DataFrame(data)

# Define a simple Bayesian Network structure

class BayesianNetwork:

def \_\_init\_\_(self):

self.model = {}

def fit(self, data):

for column in data.columns:

self.model[column] = data[column].value\_counts(normalize=True).to\_dict()

def predict(self, evidence):

probabilities = {}

for column, value in evidence.items():

if column in self.model and value in self.model[column]:

probabilities[column] = self.model[column][value]

else:

probabilities[column] = 0

return probabilities

# Create and fit the model

model = BayesianNetwork()

model.fit(heart\_disease)

# Example query: Predict heart disease given some evidence

evidence = {

'age': 63,

'sex': 1,

'cp': 3,

'trestbps': 145,

'chol': 233,

'fbs': 1,

'restecg': 0,

'thalach': 150,

'exang': 0,

'oldpeak': 2.3,

'slope': 0,

'ca': 0,

'thal': 1

}

# Predict probabilities

probabilities = model.predict(evidence)

print(probabilities)

OUTPUT:-

