

GHARDA FOUNDATION
GHARDA INSTITUTE OF TECHNOLOGY, LAVEL
Department of Computer Engineering

Evaluation Sheet

Class: TE-Computer Engineering

Sem: VI

Subject: **Artificial Intelligence Lab(CSL604)**

Experiment No: 10

Title of Experiment: Study the implementation Bayesian belief network in AI using python.

Name of Student: Niraj Nitin Surve

Roll No: 68

Date of Performance: 23/03/2023

Sr. No.	Evaluation Criteria	Max Marks	Marks Obtained
1	Practical Performance	8	
2	Oral	5	
3	Timely Submission	2	
	Total	15	

Signature of Subject Teacher
(Prof. M. A. Khandke)

Program Code –

```
# Import the necessary libraries
from pgmpy.models import BayesianNetwork
from pgmpy.factors.discrete import TabularCPD
import networkx as nx
import matplotlib.pyplot as plt

# Define the structure of the Bayesian network
model = BayesianNetwork([('A', 'B'), ('C', 'B'), ('B', 'D')])

# Define the conditional probability tables (CPDs) for each node
cpd_a = TabularCPD(variable='A', variable_card=2, values=[[0.6], [0.4]])
cpd_c = TabularCPD(variable='C', variable_card=2, values=[[0.7], [0.3]])
cpd_b = TabularCPD(variable='B', variable_card=2,
                    values=[[0.9, 0.8, 0.5, 0.3], [0.1, 0.2, 0.5, 0.7]],
                    evidence=['A', 'C'], evidence_card=[2, 2])
cpd_d = TabularCPD(variable='D', variable_card=2,
                    values=[[0.6, 0.1], [0.4, 0.9]],
                    evidence=['B'], evidence_card=[2])

model.add_cpds(cpd_a, cpd_c, cpd_b, cpd_d)
model.check_model()

# Define the node positions
pos = {'A': (0, 0), 'C': (1, 0), 'B': (0.5, 0.5), 'D': (1.5, 0.5)}

# Plot the graph of the model
nx.draw(model, pos=pos, with_labels=True)
plt.show()
```

Output –

