Assignment

Objective: Bayesian inference on the given data.

Problem: Bayesian network is a directed acyclic graphical representation of a set of variables and their conditional dependencies. Each variable is represented as a node in the graph and a directed edge between the nodes represents the parent-child relationship between the considered nodes. In this assignment we will estimate probability distributions or parameters of a given network. For this, we will make use of a dataset containing samples of values observed for different variables.

Input

- Line 1: n : no. of variable or nodes (N_1 , N_2 , ..., N_n)
- Line 2 to Line n + 1: Comma separated list of all possible values of variables N_1 to N_n
- Line n + 2 to Line 2n + 1: n × n matrix of 1's and 0's representing conditional dependencies, e.g. a value 1 at location (3,2) shows that N_2 is conditionally dependent on N_3
- Line 2n + 2: m : no. of samples
- Line 2n + 3 to Line 2n + 2 + m: Comma separated values observed for all variables (N_1 , N_2 , ..., N_n)

for each sample.

Sample input:

```
3
TRUE, FALSE
TRUE, FALSE
TRUE, FALSE
0 0 1
0 0 1
0 0 0
100
TRUE, FALSE, TRUE
FALSE, TRUE, FALSE
.
.
.
.
```

There are three binary variables (N_1 , N_2 , N_3) in this Bayesian network where N_3 is conditionally dependent on N_1 and N_2 . In other words, N_1 and N_2 are the parents of N_3.

Output

Your program should learn the parameters (probability distributions of each variable) of the given network and return them in the following format:

Output format: Print n lines where Line 1 will contain probability distribution of variable N_1, Line 2 will contain probability distribution of variable N_2 and so on. Round off the probability value upto 4 decimal places.

For the above problem the output is

```
0.2 0.8
0.4 0.6
0.2 0.4 0.3 0.5 0.8 0.6 0.7 0.5
```

This implies $P(N_1=TRUE) = 0.2$, and $P(N_1 =FALSE) = 0.8$.
Similarly $P(N_2 =TRUE) = 0.4$, and $P(N_2 =FALSE)= 0.6$.
Further, $P(N_3=TRUE|N_1 =TRUE, N_2 =TRUE) = 0.2$, $P(N_3 =TRUE|N_1 =TRUE, N_2 =FALSE) = 0.4$, $P(N_3=TRUE|N_1 =FALSE, N_2 =TRUE) = 0.3$ and so on.

Important instructions:

1. The assignment to be done individually.
2. You cannot use built-in libraries/implementations for learning the parameters of the Bayesian network.
3. Please do not search the Web for solutions to the problem. Your submission will be checked for plagiarism with the codes available on the Web as well as the codes submitted by others. Anyone found guilty will be awarded a suitable penalty as per IIT rules.
4. Your code will be automatically evaluated. You get a zero if your output is not automatically parsable.
5. You are allowed to use only Python programming language.

What to submit:
1. In google-classroom submit a file named <RollNo>.py: Source code, your solution to the problem developed using Python.
2. Submit your code on the hackerrank platform. You need to sign up on hackerrank using your iitj e-mail id. Hackerrank link: https://www.hackerrank.com/bayesian-inference

Evaluation:

Your submission will be auto-graded. This means that it is absolutely essential to make sure that your code follows the input/output specifications of the assignment. Note that the performance of your method will not be disclosed until the late submission deadline.

Late submission deadline and penalty: After the deadline, maximum achievable marks will be reduced by 50%. It means if you submit after 2 days from the deadline, zero marks will be awarded. This also applies to the re-submissions which are done past the Submission deadline.