Assignment 3: Bayesian Network

1. Introduction

The goal of this assignment is to implement Bayes Net and make inference: both exact inference and approximate inference. The initial code contains the following files, available in the zip file `bayesnets.zip`.

Files you will edit:	
`bayesianNetwork.py`	All functions that you need to modify in this file. You also need to submit this file for grading.
Files supporting:	
`main.py`	This is the main file to run this assignment.
`models/`	This is the folder containing Bayes Nets model description files
`testcases/`	This is the folder containing query files

Command

```
python main.py --model=<file model> --testcase=<file test>
```

Example 1:

```
python main.py --model=model01.txt --testcase=testcase01.txt
```

2. Your task

In this assignment, your task is to construct the BayesianNetwork class, implement its methods to describe Bayes Nets and inference on Bayes Nets including exact inference and approximate inference.

- Implement exact inference is mandatory part.
- Implement approximate inference is optional and bonus part.

In particular, you need to complete the following method in the class `BayesianNetwork`:

1. Complete method `init()` to initialize an object of the class `BayesianNetwork`. This method receives an input as a file containing the description of Bayes Nets. The detail of its format is described in **Section 3.1**.
2. Complete method `exact_inference()` to inference exactly on Bayes Nets. This medhod receive an input as a file containing the description of a query. The detail of its format is described in **Section 3.2**.
3. Complete method `approx_inference()` to inference approximately on Bayes Nets. This method receives an input as a file containing the description of a query. The detail of its format is described in **Section 3.2**.

Note that: You are allowed to write other supporting methods in the class `BayesianNetwork` or write other supporting classes but all of them MUST be in bayesianNetwork.py

3. File format

3.1. Format of model description file

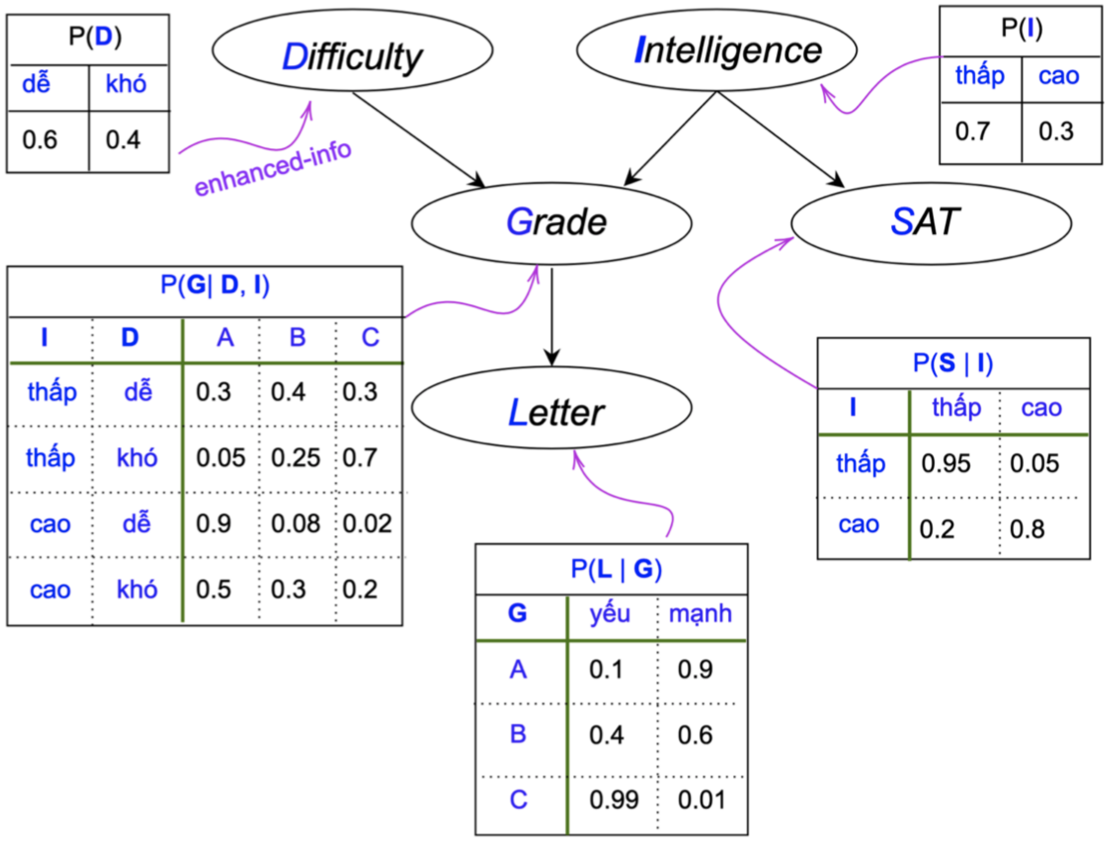
A model description file is a text file with the format as following (you can refer to Example 2 and **Figure 1** for more detail):

- The first line contains a positive integer number N (2≤N≤20). It indicates the number of variables in the Bayes Net (or the number of nodes on DAG graph shown in **Figure 1**).
- From the second line, each line describes a node in DAG graph and listed in the topological order. Each line contains the necessary information for each node in the DAG graph which contains four parts, including: **NODE; PARENTS; DOMAIN; SHAPE; PROBABILITIES** and is separated by semicolon **‘;’**

- **NODE**: is the name of the random variable (or the name of current node). The name is a string of letters or numbers, and does not contain special characters (e.g. &, %, $).
- **PARENTS**: is the list of its parents and separated by colons **‘,’**. The order of these parent nodes is matter since it is used to determine the proper probability values PROBABILITIES. This list can be empty when the current node does not have any parents.
- **DOMAIN**: contains a list of all possible values for the current node (e.g. Thap, Cao). These values are separated by colons **‘,’**.
- **SHAPE**: is the shape of the multidimensional array $D_1 \times D_2 \times .. \times D_m$ where $D_{1, \ldots,} D_{m-1}$ is the corresponding domain size of the parent nodes (listed in the same order of PARENTS). D_m is the domain size of the current node.
- **PROBABILITIES**: is a multidimensional array of $D_1 \times D_2 \times \ldots \times D_m$ real numbers stored in row-major order like C/C++. This array contains the conditional probability table (CPT) associated with the current node.

Example 2:

```
5
I;;Thap,Cao;2;0.7,0.3
D;;De,Kho;2;0.6,0.4
G;I,D;A,B,C;2,2,3;0.3,0.4,0.3,0.05,0.25,0.7,0.9,0.08,0.02,0.5,0.3,0.2
L;G;Yeu,Manh;3,2;0.1,0.9,0.4,0.6,0.99,0.01
S;I;Thap,Cao;2,2;0.95;0.05,0.2,0.8
```



P(D)	
dễ	khó
0.6	0.4

P(I)	
thấp	cao
0.7	0.3

P(G| D, I)				
I	D	A	B	C
thấp	dễ	0.3	0.4	0.3
thấp	khó	0.05	0.25	0.7
cao	dễ	0.9	0.08	0.02
cao	khó	0.5	0.3	0.2

P(S | I)		
I	thấp	cao
thấp	0.95	0.05
cao	0.2	0.8

P(L | G)		
G	yếu	mạnh
A	0.1	0.9
B	0.4	0.6
C	0.99	0.01

Figure 1: A Bayes Net.

Note that: The initial code already handled reading this format. You DO NOT need to handle any cases that the model description file is not in the right format described as above.

3.2. Format of query file

Each query file contains only ONE query. The query contains two parts separated by the semicolon **‘;’**:

- Part 1 is the list of query variables and separated by colons **‘,’**. These variables are assigned specific values in their domain through assignment character **‘=’**. Note that this list is not empty (as shown in **Example 3** and **4**).

- Part 2 is the list of evidence variables and separated by colons **‘,’**. These variables are assigned specific values in their domain through assignment character **‘=’**. Note that this list can be empty (as shown in **Example 3** and **4**).

Example 3:

```
G=A;I=Cao,D=Kho
```

It is equivalent to compute P(G=A | I=Cao, D=Kho)

Example 4:

```
G=B,S=Cao;
```

It is equivalent to compute P(G=B, S=Cao)

Note that: The initial code already handled reading this format. You DO NOT need to handle any cases that the model description file is not in the right format described as above.

4. Requirements and score

4.1. Requirements:

- Maximum time for each query: 20 second
- Allowed error range: ± 10^-5
- Environment: python 3.7
- Supporting libraries: numpy, and other standard libraries. **Please ask if you are not sure about the library you are using.**

4.2. Score:

The total score of this assignment is 110 points, including: 100 points for exact inference and 10 bonus points for approximate inference.

- There are 25 exact inference queries (4 points for each).
- There are 5 approximate inference queries (2 points for each).