**Question 1: Python**

1. Construct a Bayesian Belief Network for the given data.

Use appropriate methods to predict the following:

1. Predict the probability of a patient having the following properties: Age is 22, first delivery with delivery time 2 and with no heart problem or blood pressure and having normal delivery
2. Predict the type of delivery for the lady with the following characteristics: Age is 26, admitted for second delivery with time of delivery 0, having blood pressure and no heart problem.
3. Infer the probability for the data: Age: 36, 4th delivery, with time of delivery 1 and has blood pressure and no heart problem to have normal delivery.

**Notes on question 1:**

**1. Introduction to Bayesian Belief Network:**

To construct Bayesian Belief Network on the given data, we are using **BayesianNetwork** method from pomegranate package.

**Pomegranate** is a Python package that implements fast and flexible probabilistic models ranging from individual probability distributions to compositional models such as Bayesian networks and hidden Markov models.

**Bayesian Network Model**

A Bayesian network is a directed graph where nodes represent variables, edges represent conditional dependencies of the children on their parents, and the lack of an edge represents a conditional independence.

**Bayesian networks can be initialized in two ways, depending on whether the underlying graphical structure is known or not**:

(1) the graphical structure can be built one node at a time with pre-initialized distributions set for each node,

or (2) both the graphical structure and distributions can be learned directly from data.

We are using the second way to initialize our Bayesian networks.

**2. Code walkthrough:**

In our code, **BayesianNetwork.from\_samples(dataset, algorithm='exact')** is used to learn the structure of the network from data.

**dataset** - The data to fit the structure too, where each row is a sample and each column corresponds to the associated variable.

**algorithm** - The algorithm to use for learning the Bayesian network.

Default is ‘greedy’ that greedily attempts to find the best structure, and frequently can identify the optimal structure.

We are using ‘exact’ which uses DP/A\* to find the optimal Bayesian network.

The output of this method is the **model (The learned BayesianNetwork)**.

Using probability method on this model we have

1. Predicted the probability of a patient having the following properties: Age is 22, first delivery with delivery time 2 and with no heart problem or blood pressure and having normal delivery.

2. Infer the probability for the data: Age: 36, 4th delivery, with time of delivery 1 and has blood pressure and no heart problem to have normal delivery.

Using predict method on this model, we have predicted the type of delivery for the lady with the following characteristics: Age is 26, admitted for second delivery with time of delivery 0, having blood pressure and no heart problem.

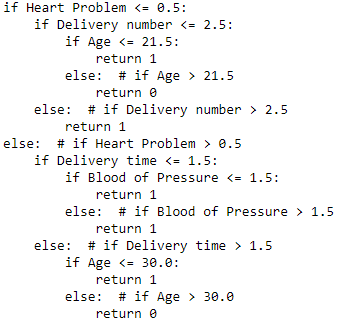
**Since the prediction is coming out as '0', hence the class value will be "No" which means the lady will have a normal delivery.**

**Question 2: Prolog**

1. Use Any of the decision tree algorithms to build a decision tree for the given data
2. Create rules from the decision tree.
3. Code the rules into a Prolog Knowledge base
4. Get user data as input and give the type of delivery as output.

**Notes on question 2:**

1. We have used python to generate decision tree using ginni index.
2. The algorithm we have used to derive rules (in the form of if-else) from the decision tree is as below:



1. We have fed the above rules to Prolog.

The snapshot of the prolog program along with its output:

