Lab 3
Bayesian Networks

Ex. 1 (1p) A system for classifying emails uses a Bayesian network to evaluate the probability that an email is spam (S) based on certain observed features. The network includes the following variables:

- S: The email can be spam ($S = 1$) or non-spam ($S = 0$).
- O: The email may contain the word "offer" ($O = 1$) or not ($O = 0$).
- L: The email may contain links ($L = 1$) or not ($L = 0$).
- M: The email may be long ($M = 1$) or not ($M = 0$).

The structure of the Bayesian network:

- Spam (S) influences the probability that the email contains the word "offer" (O) and that it contains links (L).
- The length of the email (M) is influenced both by whether it is spam (S) and by the presence of links (L).

(*Conditional*) *Probability Distributions*:

- $P(S = 1) = 0.4, \quad P(S = 0) = 0.6$
- $P(O = 1|S = 1) = 0.7, \quad P(O = 1|S = 0) = 0.1$
- $P(L = 1|S = 1) = 0.8, \quad P(L = 1|S = 0) = 0.3$
- $P(M = 1|S = 1, L = 1) = 0.9, \quad P(M = 1|S = 1, L = 0) = 0.5$
- $P(M = 1|S = 0, L = 1) = 0.6, \quad P(M = 1|S = 0, L = 0) = 0.2$

Requirements (using pgmpy):

a) (0.5p) Identify the independencies in the network.
b) (0.5p) Determine how the Bayesian network classifies emails based on the attributes O, L, and M.

Ex. 2 (0.5p) Solve exercise 1.b) from the previous lab (Lab 2) by using a Bayesian network and compare the results.

Ex. 3 (1.5p) A game between two players, P_0 and P_1, unfolds as follows:

- A (fair) coin is tossed first to decide who starts: P_0 or P_1;
- In the first round, the designated player rolls their own die; let n be the number obtained;
- In the second round, the other player flips their own coin $2n$ times; let m be the number of heads obtained.

The player from the first round wins if $n \geq m$, otherwise the second-round player wins. We also know that player P_1 is *dishonest*, having brought a *rigged coin* with a probability of getting heads equal to 4/7. In contrast, P_0's *coin* is *fair*, and both *dice* are *fair* as well.

1. (0.5p) Estimate which of the two players has the higher chance of winning by simulating the game 10000 times.
2. (0.5p) Using pgmpy, define a Bayesian network that describes the context above.
3. (0.5p) Using the model above, determine who is most likely to have started the game, knowing that *only one head* was obtained in the second round.