Due Date: Wednesday, November 20 by 23:59 (11:59pm).
**Submission Instructions:**

- Assignments are to be submitted through LEARN, in the Dropbox labelled *Assignment 3 Submissions* in the Assignment 3 folder.

- Late assignments will be accepted up until November 22, 2019 at 23:59. Please read the course policy on assignments submitted after the official due date. *No assignment will be accepted, for any reason after November 22 at 23:59.*

- The following exercises are to be done individually.

- You should submit **TWO** files for the assignment.

    - **writeup.pdf**
        * Include your name, your uwaterloo email address, and your student number.
        * If you handwrite your answers, make sure your handwriting is legible and the pictures or scan is clear. You may get a mark of 0 if we can not read your handwriting.

    - **code.zip**
        * Include your programs, scripts to run your programs, and README.txt file with instructions to run the script. You may get a mark of 0 if we cannot run your program. Ensure that your code is **well documented**. You may lose points if we can not understand your documentation. **Note that for this assignment you are not required to do any programming. However, I will also note that writing code will likely be much easier than doing the calculations by hand.**

Lead TA: Charupriya Sharma (c9sharma@uwaterloo.ca). Office hours Wednesdays from 14:30-16:00 in DC3131.

Question 1 (15 points)

Consider the following Bayes net (we leave out the Conditional Probability Tables).
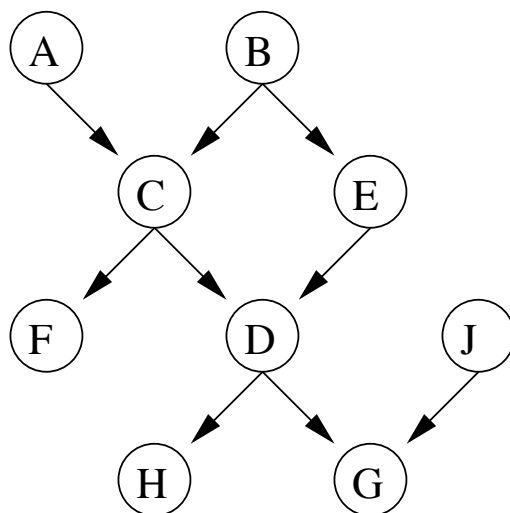
For each of the questions below, state whether the independence relation holds. Give a justification for your answer using the d-separation criterion: if d-separation holds, state why each *undirected* path is blocked; if it does not hold, describe one undirected path which is unblocked. (Note: there are no more than two undirected paths between any pair of nodes).

1. A and B are independent

2. A and B are independent given D

3. A and E are independent

4. A and E are independent given C

5. A and E are independent given B

6. A and E are independent given B and C

7. A and E are independent given D

8. A and E are independent given F

9. F and H are independent

10. J and E are independent

11. J and E are independent given G

12. J and E are independent given A

13. G and A are independent

14. G and A are independent given C

15. G and A are independent given C and D

Question 2. Variable Elimination (0 points)

This part of the assignment is worth zero points. However, your implementation will be used Question 3. *If you do the implementation then upload it with your assignment.*

Implement the variable elimination algorithm by coding the following four functions in the programming language of your choice.

a. restrictedFactor=restrict(factor, variable, value): a function that restricts a variable to some value in a given factor

b. productFactor = multiply(factor1, factor2): a function that multiplies two factors

c. resultFactor = sumout(factor, variable): a function that sums out a variable given a factor

d. resultFactor = inference(factorList, queryVariables, orderedListOfHiddenVariables, evidenceList): a function that computes *Pr(queryVariable | evidenceList)* by variable elimination. This function should restrict the factors in *factorList* according to the evidence in *evidenceList*. Next, it should sum out the hidden variables from the product of the factors in *factorList*. The variables should be summed out in the order given in *orderedListOfHiddenVariables*. Finally, the answer should be normalized. Note that you might want to implement an additional function **normalize(factor)** to help you do this.

Here are some useful tips for this part of the assignment.

Tip Factors are essentially multi-dimensional arrays. Therefore, you may want to use this data structure. However, you are free to use any data structure that you want.

Tip Test each function individually using the simple examples we covered in class. Debugging the entire variable elimination algorithm at once can be tricky.

Question 3 Bayes Nets (80 points)

Your dog Fido has been howling for the last three hours and you want to decide whether or not to take him to the vet, or just put in earplugs and go back to sleep. You have the following information available to you:

- You know that Fido often howls if he is genuinely sick. However, he is a healthy dog and is only sick 5% of the time. If Fido is really sick he probably has not eaten much of his dinner and has food left in his bowl. In the past you have observed that when Fido is sick then 60% of the time he does not eat. However, about 10% of the time, when Fido is healthy he still does not eat his dinner.

- You also know that Fido often howls is there is a full moon or the neighbour's dog howls. The neighbour's dog sometimes howls at the full moon and sometimes howls when your neighbour is away. However, the neighbour's dog is never affected by Fido's howls. You know that (since you live on Earth) there is a full moon once out of every twenty-eight days. You have observed that your neighbour travels three days out of every ten. You also know that if there is no full moon and the neighbour is at home then the dog never howls, but if there is no full moon and the neighbour is away then the dog howls with probability 0.5. If there is a full moon then the dog is more likely to howl; if the neighbour is at home then it howls with probability 0.4, but if the neighbour is also away then the probability of howling increases to 0.8.

- Finally, you know that if all the triggers are there (i.e. full moon, howling neighbour's dog, Fido sick) then Fido will howl with probability 0.99, but if none of the triggers are there, then Fido will not howl. If Fido is sick then he is likely to howl. In particular, if he is sick (but there are no other triggers) then Fido will howl with probability 0.5. If Fido is sick and there is another trigger then he is even more likely to howl – if the neighbour's dog is also howling then Fido will howl $\frac{3}{4}$'s of the time, while if there is a full moon then Fido howls 90% of the time. If Fido is *not* sick then he is less likely to howl. The full moon and the neighbour's dog will only cause him to howl with probability 0.65, while if there is only a full moon then he will howl 40% of the time, and if there is no full moon, but the neighbour's dog is making noise, then he howl's with probability 0.2.

1. (32 points) Given the information about Fido, construct a Bayes Network. Show the graph and the conditional probability tables. The network should encode all the information stated above. It should contain six nodes corresponding to the following binary random variables:

   - *FH* – Fido howls
   - *FS* - Fido is sick
   - *FB* - there is food left in Fido's food bowl
   - *FM* - there is a full moom
   - *NA* - the neighbour is away
   - *NDG* - the neighbour's dog howls

The edges in your Bayes Network should accurately capture the probabilistic dependencies between these variables.

For the next set of questions indicate what queries (i.e. Pr(vars | evidence)) you used to compute the probability. Whether you answer the queries by hand or by using your code, provide a printout of the factors computed at each step of variable elimination. If you think that some of the probabilities will not change, then you do not have to redo the calculations, but you do need to provide an explanation.

2. (12 points) What is the prior probability that Fido will howl (i.e. Pr(FH))?

3. (12 points) You can hear Fido howling, and you are concerned that he is sick. You look out the window and see that the moon is full. What is probability that Fido is sick?

4. (12 points) You next walk to the kitchen to see if there is any food left in Fido's bowl. You note that the bowl is full – that is Fido has not eaten. How does your belief change about Fido being sick now that you know there is a full moon and Fido has not eaten?

5. (12 points) Finally, you decide to call your neighbour to see if they are home or not. The phone rings and rings so you conclude that your neighbour is away. Now, given this information, how does you belief about Fido being sick change?

Question 4 (30 pts)
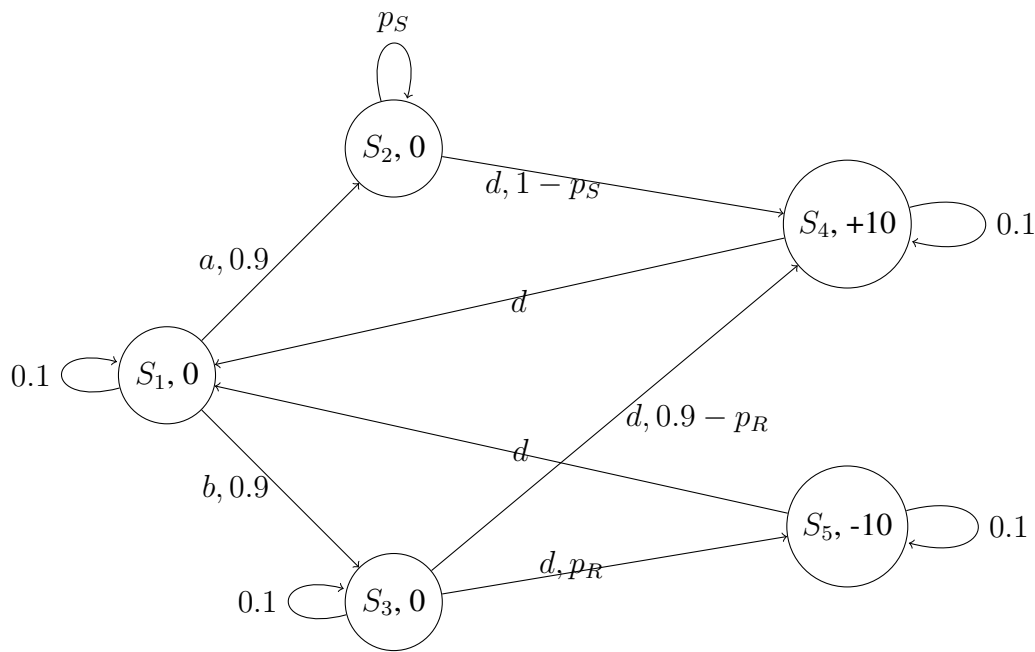
Consider the MDP shown in the above figure. There are five states $(S_1, S_2, S_3, S_4, S_5)$. The reward in states S_1, S_2 and S_3 is equal to 0, while the reward for state S_4 is 10 and the reward for state S_5 is -10. In state S_1 there are two possible actions a and b, while in all other states there is a single action d. Each action is stochastic. With probability 0.1 the result of taking some action is a self-transition (i.e. the agent remains in the same state) while with probability 0.9 the next state is indicated in the diagram above. There are two exceptions:

- State S_2 is a *sticky* state. It has probability of p_S (the stickiness factor) of self-transition, and probability of $1 - p_S$ of transitioning to state S_4.

- State S_3 is a *risky* state. The probability of self-transition is 0.1. However, with probability p_R (the riskiness factor) the agent will move to state S_5 while with probability $0.9 - p_R$ the agent will transition to state S_4.

Assume that we are interested in total discounted reward and that the discount factor is $\gamma = 0.95$.

1. Set $p_S = 0.2$. What is the value function and the optimal policy for this MDP if the risk probability is $p_R = 0.01$? What is the optimal value function and policy if the risk probability is $p_R = 0.03$? For what value of p_R is the agent indifferent between taking action a in state S_1 or action b in state S_1? (You can assume that the agent starts out in state S_1 if you wish.)

2. Now increase the stickiness probability to $p_S = 0.6$. Repeat the previous questions for $p_R = 0.1$ and $p_R = 0.2$. Again, what is the indifference level for p_R? That is, for what value of p_R is the agent now indifferent between taking action a and b in state S_1?