The following exercises are intended to further your understanding of Probability, Belief Nets, Decision Nets (simple decisions), Sequential Decision and Game Theory. 
(From Chapters 13, 14, 15, 16, 17)

Submission: You should hand-in hardcopies of Problems 1 to 5. For Problem 6: see below.

---

**Problem 1** [5 points]  Simple Probabilistic Inference

**a [3]: Bayes Rule**
Suppose Box#1 contains 20 candies and 10 fruits, and Box#2 contains 7 candies and 5 fruits. One of the boxes is chosen at random (with equal probability) and an item is selected from the box and found to be an candy. Find the probability that the candy came from Box#1.

*Hint: Use Bayes’ theorem!*

**b [2]: Paradox?**
Harry and Dursley each practice casting spells for two sessions. In the first session, over 100 attempts, Harry’s spells work correctly 60 times, while Dursley’s are correct 90 times. In the second session, Harry is accurate just 1 time in 10, while Dursley is correct 300 times in 1000. Notice Harry is less accurate than Dursley in both individual sessions (session1: 0.6 is less 0.9; and session2: 0.1 is less 0.3).

What are their respective accuracies when considering both sessions together? Does this seem strange to you?

---

**Problem 2** [10 points]  Conditional Dependence

Given sets of variables \( W, X, Y \), say \( D(W, X | Y) \) if \( W \) is dependent on \( X \), given \( Y \) — i.e.,

\[
\exists w, y, x \quad P(W = w | Y = y) \neq P(W = w | Y = y, X = x).
\]

Also \( D(W, X | \{Y, Z\}) \) iff \( \exists w, x, y, z \quad P(W = w | Y = y, Z = z) \neq P(W = w | Y = y, Z = z, X = x) \).

(Note \( D(W, X | \{Y, Z\}) \) \( \iff \neg I(W, X | \{Y, Z\}) \).

Prove or disprove the following probabilistic dependency claims.

**a [5]:** 
\[
D(W, \{X, Z\} | Y) \implies D(W, Z | \{Y, X\}) \lor D(W, X | Y).
\]

**b [5]:** 
\[
D(W, X | \{Y, Z\}) \implies D(W, X | Y) \lor D(W, X | Z).
\]

---

**Problem 3** [20 points]  Hidden Markov Model

You are playing a dice game in the *Hogwart’s Casino*, against Malfoy. Unfortunate, Malfoy cheats... when he can, he jinxes the single dice to change from being fair (with distribution
(1) to rigged (r), based on the distributions

| v  | P(v | D = f) | P(v | D = r) |
|----|-----------|------------|
| 1  | 1/6       | 0.80       |
| 2  | 1/6       | 0.04       |
| 3  | 1/6       | 0.04       |
| 4  | 1/6       | 0.04       |
| 5  | 1/6       | 0.04       |
| 6  | 1/6       | 0.04       |

where in general the variable D (later D_t) refers to the current state of the die, which is a hidden variable. As he needs to cast the jinx undetected, which is difficult, he only does this occasionally — only 10% of the time. Fortunately for fair play, Dobby can counterjinx it back (from r to f). However, he also does not want to be caught, and so can only change the die back 10% of the time. So we have

\[
\begin{align*}
P(D_{t+1} = r | D_t = f) &= 0.1 \\
P(D_{t+1} = f | D_t = f) &= 0.9 \\
P(D_{t+1} = f | D_t = r) &= 0.1 \\
P(D_{t+1} = r | D_t = r) &= 0.9
\end{align*}
\]

(2)

Now imagine you observe the sequence of die rolls:

\[
\vec{S} = \langle 4, 1, 2, 3, 1, 3, 1, 5, 6 \rangle
\]

Let \(\vec{S}_{ij}\) be the subsequence between roll #i and roll #j (inclusive), so \(\vec{S}_{1:3} = \langle 4, 1, 2 \rangle\) and \(\vec{S}_{1:10} = \vec{S}\). You may assume the \{f, r\} are the only two possible states of the die, and that the initial “jinx-state” of the die is 50/50 — \(P(D_0 = r) = 0.5\).

a [5]: Use simple filtering to compute the probabilities \(P(D_t = r | \vec{S}_{1:t})\) for \(t = 1..10\).

b [5]: Now you get to use smoothing; compute \(P(D_t = r | \vec{S}_{1:10})\) for \(t = 1..10\).

c [5]: What is the most likely interpretation \(\text{argmax}_{\vec{d}} P(\vec{D} = \vec{d} | \vec{S}_{1:10})\) where \(\vec{D}\) is shorthand for \(\langle D_1, \ldots, D_{10} \rangle\)?

d [5]: If Lucius is watching, Malfoy is more likely to try to show-off and Dobby is less able to change the die back, meaning the transition probabilities will be:

\[
\begin{align*}
P_{+L}(D_{t+1} = r | D_t = f) &= 0.2 \\
P_{+L}(D_{t+1} = f | D_t = f) &= 0.8 \\
P_{+L}(D_{t+1} = r | D_t = r) &= 0.95 \\
P_{+L}(D_{t+1} = f | D_t = r) &= 0.05
\end{align*}
\]

(3)

which is different from the \(P(D_{t+1} | D_t)\) values presented in Equation 2. (You should view those values as a shorthand for \(P_{-L}(D_{t+1} | D_t)\).) Note that Equation 1 is unaffected.

Do you think Lucius is watching? That is, is \(P_{+L}(\vec{D} = \vec{d} | \vec{S}) > P_{-L}(\vec{D} = \vec{d} | \vec{S})\)?

Problem 4 [30 points] Best Single Action, Conditioning Events
Homer (a very annoying person) shows you three sealed envelopes, and tells us exactly one
contains an admission to Krusty’s show $K$ (which is good), while the other two are passes to see Itchy-&-Scratchy $I&S$ (which is bad). You select one; call it $A$. Before you open $A$, Homer opens one of the unpicked envelopes ($B$), and shows you that it is an $I&S$ pass. He then gives you a choice: either keep your letter $A$, or exchange it for the other unopened letter, $C$.

a [10]: What should you do — open your original $A$ or instead switch, and open $C$? For now, you should assume that Homer always shows an $I&S$ letter and gives this option of switching.

[Hint: Feel free to implement a simulation, to verify your answer.]

b [10]: Write a decision net that encodes this situation. You will need to specify the chance-nodes, action-nodes and value-nodes, the arcs connecting them, and the parameters. You should include only the arcs that are necessary. Here, you should assign the value 1 to getting a Krusty letter, and the value 0 for getting the $I&S$ pass.

Please write a paragraph explaining your decision net — specifying how it works, motivating why you connected $x$ to $y$ (or not), etc. This explanation is worth as much as the decision net itself.

c [5]: Now imagine Homer began with 100 envelopes, only one of which contains the Krusty invitation. After you pick $A$, Homer shows you one of the non-Krusty letters, say $B$, then gives you option of keeping your $A$, or randomly selecting one of the remaining 98 letters. Now what should you — keep $A$ or draw another envelope?

d [5]: Now suppose Homer does not always give the participant the opportunity to switching. How does this change the problem?

[Hint: Think of why Homer might, or might not, offer you the option of selecting a different letter.]

Problem 5 [40 points] Actions — Markovian Property . . . Game Theory

It’s soccer time!

Player F is approaching the goal area on a breakaway — it’s just him versus the goalie, J. As he approaches, he sees J at the centre of the goal area. F now has to decide whether he should

1. kick to the east, “pe”
2. kick to the west, “pw”
3. dribble the ball to get closer, “d”.

Once closer, he can then shoot east “pe$_2$” or west “pw$_2$”.

What should he do?

a [5]: For now, assume a simple transition function:

\[
\begin{align*}
P(S \mid T_0, pe) &= 0.3 & P(B \mid T_0, pe) &= 0.7 \\
P(S \mid T_0, pw) &= 0.8 & P(B \mid T_0, pw) &= 0.2 \\
P(T_1 \mid T_0, d) &= 0.9 & P(B \mid T_0, d) &= 0.1 \\
P(S \mid T_1, pe_2) &= 0.9 & P(B \mid T_1, pe_2) &= 0.1 \\
P(S \mid T_1, pw_2) &= 0.2 & P(B \mid T_1, pw_2) &= 0.8
\end{align*}
\]
where \( T_0 \) is the initial state, \( T_1 \) is the state where the player F is closer to the goal, \( S \) is “scoring a goal” and \( B \) corresponds to not scoring a goal (perhaps because J blocked the shot, or smothered the ball).\(^1\)

The payoff here depends only on the state

\[
\begin{align*}
R_F(T_0) &= 0 \\
R_F(T_1) &= 0 \\
R_F(S) &= 1 \\
R_F(B) &= -1
\end{align*}
\]  

(The subscript refers to the player “F”.)

What should F do in \( T_0 \): shoot west, shoot east or dribble...and if F dribbles, should he then shoot west or east?

[Hint: Compute \( E[pe|T_0] \), the expected payoff if F performs \( pe \) in situation \( T_0 \), etc.]

When F follows this optimal policy, what is his expected payoff?

b [2]: A “policy” assigns a single action to each state. We can allow F to perform actions \emph{stochastically}: e.g., in state \( T_0 \), F could

\begin{itemize}
  \item shoot east with probability 0.3
  \item shoot west with probability 0.0
  \item dribble with probability 0.7
\end{itemize}

or whatever.

What is F’s best stochastic distribution of actions?

c [5]: Now let’s consider the goalie: what should she do? In the \( T_0 \) configuration, suppose the (possibly different) goalie \( G \) can stop 80% of the kicks by the (possible different) opponent \( P \) if she guesses correctly, but only 10% if she guesses wrong — i.e.,

\[
\begin{align*}
P(B | T_0, P \text{ shoots east, } G \text{ goes east}) &= 0.8 \\
P(B | T_0, P \text{ shoots west, } G \text{ goes west}) &= 0.8 \\
P(B | T_0, P \text{ shoots east, } G \text{ goes west}) &= 0.1 \\
P(B | T_0, P \text{ shoots west, } G \text{ goes east}) &= 0.1
\end{align*}
\]  

(5)

In each case, \( P(S | T_0, \ldots) = 1 - P(B | T_0, \ldots) \). (Here, we assume that P has to shoot — i.e., he does not have the option of dribbling.)

The goalie can decide whether she should go east \( ge \) or west \( gw \). If she knew which way P would shoot, her action would be obvious: \emph{E.g.}, if P always goes east, clearly \( G \) should also go east — i.e., perform \( ge \). (Here we use \( R_G(\cdot) = -R_P(\cdot) \) as the goalie is trying to avoid being scored on — hence \( R_G(S) = -1 = -R_P(S) \) and \( R_G(B) = 1 = -R_P(B) \).)

\(^1\)For now, we assume the probabilities are static — that is, neither the shooter nor the goalie will alter their respective policies in response to the other. But see below.

Also, to interpret these probabilities: J anticipates that F will shoot east in situation \( T_0 \), and so has a 70% chance of blocking an east shot. If F shoots west, however, he has a 80% chance of scoring. However, if F tries to advance the ball, there is a 10% chance of losing the ball. But if F succeeds (reaches situation \( T_1 \)), then J assumes F will shoot west. As F is closer, J has a better chance of blocking the ball if she guessed correctly (\emph{i.e.}, F goes west), but J has less chance of blocking the ball if she guessed wrong.
Unfortunately for G, she does not know which way P will go. Fortunately, G does have a stochastic model of P: she knows that P will go east 70% of the time and west the remaining 30% — i.e., \( P(pe \mid T_0) = 0.7 \) and \( P(pw \mid T_0) = 0.3 \).

What should G do? 

[Hint: Compute \( P(B \mid T_0, ge) \) and \( P(B \mid T_0, gw) \), then use this to compute expected utility, \( E[ge \mid T_0] \) and \( E[gw \mid T_0] \).]

Is this “policy” fixed or stochastic — i.e., should G perform the same action each time, or perform some stochastic action? 

d [3]: In general, we can write \( P(pe \mid T_0) = \alpha \), for some specific \( \alpha \in [0, 1] \). What should G do, as a function of this \( \alpha \)?

e [5]: How did G get this \( \alpha \)? Let’s assume that G and P have had \( N \) 1-on-1 previous encounters, and that P shot to the east \( k \) times, and \( N - k \) times to the west. Here, G would set \( \alpha = \alpha_N = k/N \), and act accordingly.

Now assume that everyone knows everything: G and P both know G’s effectiveness (Equation 5), the utility values \( R_P(\cdot) \) and \( R_G(\cdot) \), how G will set \( \alpha \), etc.

The rest of this question explores what will happen here.

To break the symmetry, assume that P went east on his first shot; this means G will set \( \alpha_1 = 1/1 = 1.0 \). Assume for now that G has to follow the \( \alpha \)-based policy given in the previous question. Given what P knows, what should he do in this situation?

f [5]: Of course, G is not that stupid: She will know what P knows, and will try to out-think him.

In general, we can imagine a \( 2 \times 2 \) payoff matrix:

\[
\begin{array}{c|cc}
 & \text{P’s action} & \\
\hline
\text{G’s action} & pe & pw \\
\hline
ge & \text{ } & \\
gw & \text{ } & \\
\end{array}
\]

where the \((i, j)\) entry corresponds to the expected reward for P (based on \( R_P(\cdot) = R_F(\cdot) \), from Equation 4) if P performs his \( i^{th} \) action and G performs her \( j^{th} \) action; hence the \((1, 2)\) position is the reward if P performs his first action (\( pe \)) and simultaneously G performs her second action (\( gw \)).

Fill out this table, based on the \( R_F \) utility (Equation 4) and G’s effectiveness (Equation 5).

g [5]: Suppose P performs \( pe \) with probability \( \alpha \in [0, 1] \) and \( pw \) with probability \( 1 - \alpha \), and (independently and simultaneously) G performs \( ge \) with probability \( \beta \in [0, 1] \) and \( gw \) with probability \( 1 - \beta \). Note that G must decide on her action at the same moment that P is deciding on his action — and so G cannot wait to see P’s action and then respond.

What is the expected outcome, \( f(\alpha, \beta) \)? (Here \( f : [0, 1] \times [0, 1] \rightarrow \mathbb{R} \) maps a pair of probability values to the reals.)

h [5]: Both P and G know this payoff matrix, and this \( f(\alpha, \beta) \) function. Given this information, what policy should P adapt? . . . what policy should G adapt? Does they correspond to fixed policies, or to stochastic ones?
[Hint: Determine the $\alpha$ and $\beta$ values that optimize the $f(\alpha, \beta)$ function...]

i [5]: Now suppose P practices certain moves a lot, and becomes able to shoot more effectively to the east.

\[
\begin{align*}
P( B \mid T_0, \text{P shoots east, G goes east} ) &= 0.3 \\
P( B \mid T_0, \text{P shoots east, G goes west} ) &= 0.05 \\
P( B \mid T_0, \text{P shoots west, G goes east} ) &= 0.1 \\
P( B \mid T_0, \text{P shoots west, G goes west} ) &= 0.8 \\
\end{align*}
\]

Now determine the best play for both G and P.

[Hint: You may want to produce a payoff matrix, an expected payoff function $g(\alpha, \beta)$, etc.]

**Problem 6 [40 points] Rock-Paper-Scissors: Implementation**

“Rock-Paper-Scissors” (aka “RoShamBo”) is a trivial game between two participants: Each participant independently selects one number from $\{0, 1, 2\}$; they then compare the results. If the selections are the same (e.g., $\langle 0, 0 \rangle$ or $\langle 2, 2 \rangle$), then this game is a draw. Otherwise... if the players chose $\langle 0, 1 \rangle$, then the player selecting 1 beats the player selecting 0; if the players chose $\langle 1, 2 \rangle$, then the player selecting 2 beats the player selecting 1; if the players chose $\langle 2, 0 \rangle$, then the player selecting 0 beats the player selecting 2.

If you know nothing of your opponent, of course, you can no better than just picking one of the numbers at random.

But what if you play this single player for a large number of games? Here, you have a chance to learn something of her habits. For example, perhaps you find she always plays “0”. Once you catch on to this, you can win consistently by always playing “1”. Or perhaps on game $G_i$ she just plays what you played on game $G_{i-1}$. Once again, this is extremely useful information!

Of course, she might just be trying to sucker you: She might be playing 0, 0, 0, ... until she see that you have “figured” out her “always-play-0” scheme — i.e., until she sees you playing 1 consistently. Then she will start playing the move that beats “1” — namely “2”.

Of course, you might have been waiting for this, and so now will be able to clobber her by playing “0”. This works unless, of course, your opponent was anticipating that you were anticipating that she was ...

So you see this can get very tricky, very quickly.

Your challenge is to write the RoShamBo player that wins the tournament, against all comers, including your colleagues. You can find the official rules in

http://www.cs.ualberta.ca/~darse/rsbpc.html

That website also has the ideas and implementations from earlier competitors. While you are encouraged to examine these, of course, your program must be significantly different. You must also give credit to any existing program whose ideas you used.

In addition to your code, you must also write up a description of your algorithm, and also indicate situations where you expect that your algorithm will succeed, and when you expect it to fail. (This write-up should be 1-2 pages, and is worth 25 of the 40 points.)
Finally, for those of you who are motivated by fame... the tournament’s organizer would like copies of your submissions, to add to the archive of RoShamBo players. **Submission details:** Use ASTEP to submit a single *.c file, whose name is your account name (eg, mine would be named “rgreiner”).

You should test your code by integrating it into [http://www.cs.ualberta.ca/~darse/rsb-ts1.c](http://www.cs.ualberta.ca/~darse/rsb-ts1.c) to confirm that it compiles (you will get 0 points if your program does not compile!) and also runs effectively. In particular, on each call, it should return one of the values, \{ 0, 1, 2 \}. This function can use the global variables “my\_history” and “opp\_history” which respectively provide the history of your previous actions and your opponent’s.