CS 181 Artificial Intelligence (Fall 2024), Midterm Exam

Instructions

- Time: 10:15–11:55am (100 minutes)
- This exam is closed-book, but you may bring one A4-size cheat sheet. Put all the study materials and electronic devices into your bag and put your bag in the front, back, or sides of the classroom.
- You can write your answers in either English or Chinese.
- Two blank pieces of paper are attached, which you can use as scratch paper. Raise your hand if you need more paper.
- For multiple choice questions:
  - □ means you should mark ALL choices that apply;
  - ○ means you should mark exactly ONE choice;
  - When marking a choice, please fill in the bubble or square COMPLETELY (e.g., ● and ■). Ambiguous answers will receive no points.
  - For each question with □ choices, you get half of the points for selecting a non-empty proper subset of the correct answers.

1 Multiple choice (10 pt)

Each question has one or more correct answers. Select all the correct answer(s). For each question, you get 0 point if you select one or more wrong answers, but you get 0.5 point if you select a non-empty proper subset of the correct answers. Fill your answers in the table below.

1	2	3	4	5

6	7	8	9	10

1. Consider a tree search problem with finite state space and all the edges have positive costs. The goal is to find a path from the start state to the goal state with the minimum cost. Which of the following statements is/are true?

   A. If the heuristic is admissible, A* search is guaranteed to find the optimal solution.
   B. $h(n) = 0$ is an admissible heuristic in this problem.
   C. Depth-first search always expands at least as many nodes as A* search with an admissible heuristic.
   D. Uniform-cost search is guaranteed to find the optimal solution.
   E. None of the above.

2. Figure 1 shows a genetic algorithm for solving a constraint satisfaction problem (CSP). About the missing parts in the figure, which ONE of the following statements is true?

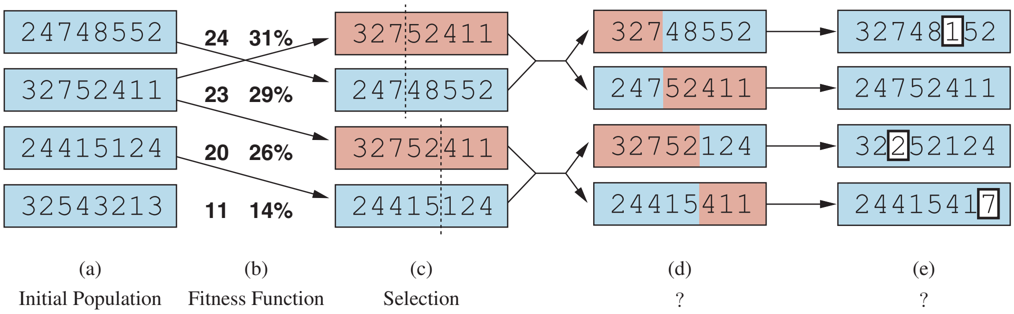


Figure 1: A genetic algorithm, illustrated for digit strings representing states.

A. (d) is a crossover operation and (e) is a mutation operation.

B. (d) is a mutation operation and (e) is a crossover operation.

C. (d) is a rotation operation and (e) is a deletion operation.

D. (d) is a deletion operation and (e) is a rotation operation.

3. Which of the following statements about game and adversarial search is/are correct?

A. In a zero-sum game with 2 player, an increase in one player's utility results in a decrease in the other player's utility.

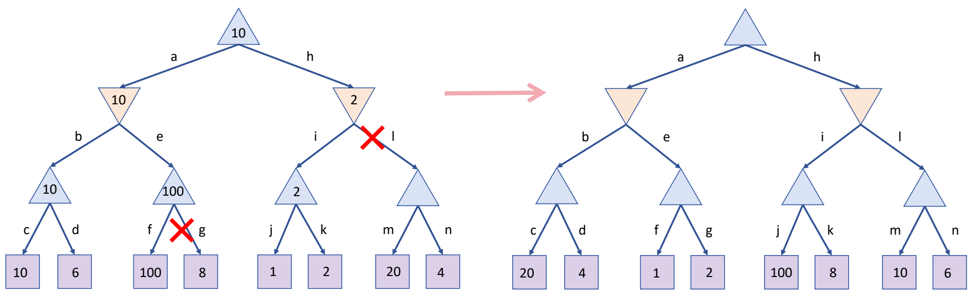
B. The primary goal of $\alpha-\beta$ pruning in minimax search is to decrease the number of nodes evaluated by skipping unneeded branches.

C. Unlike minimax search, expectimax assumes that the opponent is adversarial.

D. The time complexity of minimax is $O(b \cdot d)$ and the space complexity is $O(b^d)$, where d is the depth of the tree and b is the branching factor.

E. None of the above.

4. We are performing $\alpha-\beta$ pruning, expanding successors from left to right (traversal order: a to n), on the game trees shown below. In the left, recall the example in class, we will have 2 branches to prune. Now we change the order of the value in the leaf nodes (the right part), how will the number of branches to be pruned change w.r.t the original setting (the left part) if we expand with the same traverse order?



A. Greater than 2

B. Less than 2

C. Equal to 2

D. We cannot determine with current information.

5. Which of the following statements about first-order logic is/are correct?

   A. A sentence is satisfiable if it is true in all models.

   B. Modus Ponens is sound and complete for Horn logic.

   C. The complexity of forward chaining is usually higher than backward chaining.

   D. $(A \Rightarrow B) \wedge (\neg B \Rightarrow C) \equiv \neg B \Rightarrow (\neg A \wedge C)$.

   E. None of the above.

6. Which of the following statements about first-order logic is/are correct?

   A. $> (\text{Age}(\text{Brother}(A)), \text{Age}(A))$ is an atomic sentence.

   B. $\forall x \exists y \text{ Brother}(x, y) \equiv \exists y \forall x \text{ Brother}(x, y)$.

   C. If α is entailed by any one subset of a FOL KB, it is entailed by this KB.

   D. $\forall x\ F(x) \equiv \neg(\neg\exists x\ F(x))$.

   E. None of the above.

7. Which of the following statements about FOL expression is/are correct?

   A. "Everyone is loved by someone and hated by someone." can be translated to the following FOL expression: $\forall x\ \exists y \text{ Loves}(y, x) \wedge \text{Hates}(y, x)$.

   B. "Every student at ShanghaiTech is smart." can be translated to the following FOL expression: $\forall x \text{ Student}(x) \wedge \text{At}(x, \text{ShanghaiTech}) \wedge \text{Smart}(x)$.

   C. "Cristiano is the best football player." can be translated to the following FOL expression: $\forall x \text{ Footballplayer}(\text{Cristiano}) \wedge \text{Footballplayer}(x) \Rightarrow \text{Better}(\text{Cristiano}, x)$.

   D. "Certain drugs can cause hallucinations in humans." can be translated to the following FOL expression: $\exists x \forall y \text{ Drug}(x) \wedge \text{Human}(y) \Rightarrow$ Hallucination(x, y).

   E. None of the above.

8. Given the factors $P(A|C)$ and $P(B|A, C)$ what is the resulting factor after joining over A and summing over A?

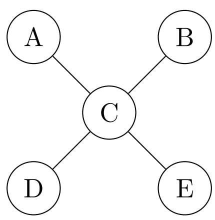
   A. $P(B)$.

   B. $P(B|C)$.

   C. $P(B, C)$.

   D. $P(A, B, C)$.

9. Given the following Markov Network, which of the following statements is/are correct?



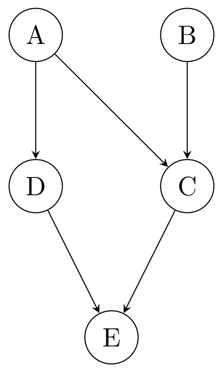
   A. $P(A|B) = P(A)$.

   B. $P(A|B, C) = P(A|C)$.

   C. $P(A, D) = P(A) \times P(D)$.

   D. None of the above.

10. Consider the Bayesian network below, which of the following statements is/are correct?



A. $P(A, B) = P(A) \times P(B)$.

B. $P(A, B \mid C) = P(A \mid C) \times P(B \mid C)$.

C. $B \perp\!\!\!\perp E | C$.

D. $A \perp\!\!\!\perp B | E$.

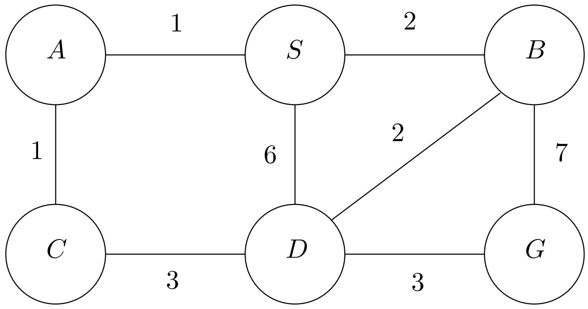
Solution:

1. ABD
2. A
3. AB
4. B
5. BCD / BD
6. A / AC
7. C / E
8. B
9. B
10. A

2 Search & CSP (10 pt)

2.1 Search (5 pt)

Consider the following search graph. S is the start state, and G is the goal state. The numbers on the edges represent the costs of the edges. All edges are bidirectional. The problem is to find the path from S to G with the minimum cost.



Recall from lecture the general algorithm for GRAPH-SEARCH reproduced below. Assume that ties are broken alphabetically (so a partial plan $S \to X \to A$ would be expanded before $S \to X \to B$).

function GRAPH-SEARCH(*problem*)
    closed $\leftarrow \emptyset$
    $fringe \leftarrow$ T()
    $fringe$.push(Node(*problem.initialState*))
    while not $fringe$.isEmpty() **do**
        node $\leftarrow fringe$.pop()
        if *problem.isGoal*(*node*.state) **then**
            return SOLUTION(*node*)
        end if
        if *node*.state $\notin$ *closed* **then**
            closed.add(*node*.state)
            for each child in *problem.expand*(*node*) **do**
                $fringe$.push(*child*)
            end for
        end if
    end while
end function

2.1.1 Data Structures (1 pt)

The data structure T is the key to the search algorithm. With the right choice of T, the algorithm can be universally optimal in terms of time complexity (Haeupler, et al., 2023). Which data structure T should be used for the fringe in GRAPH-SEARCH to implement DFS, BFS, and UCS, respectively?

- ○ DFS: stack; BFS: queue; UCS: priority queue
- ○ DFS: queue; BFS: stack; UCS: priority queue
- ○ DFS: stack; BFS: stack; UCS: queue
- ○ DFS: queue; BFS: queue; UCS: priority queue

Solution:

A

2.1.2 Admissible or Consistent (1 pt)

For the following questions, consider heuristic functions $h(n)$:

State	S	A	B	C	D	G
$h(n)$	6	5	5	4	2	0

Is $h(n)$ admissible or consistent?

- ○ $h(n)$ is admissible and consistent.
- ○ $h(n)$ is admissible but not consistent.
- ○ $h(n)$ is not admissible but consistent.
- ○ $h(n)$ is neither admissible nor consistent.

Solution:

B

2.1.3 Expanded Nodes (2 pt)

Run A* search with the heuristic function $h(n)$ on the given graph. Assume that ties are broken alphabetically. What are the nodes expanded in order?

- ○ S, A, C, D, G
- ○ S, A, B, D, G
- ○ S, B, D, G
- ○ S, A, C, B, D, G

Solution:

D

2.1.4 Optimal Path (1 pt)

Run A* search with the heuristic function $h(n)$ on the given graph. Does it return the optimal path from S to G?

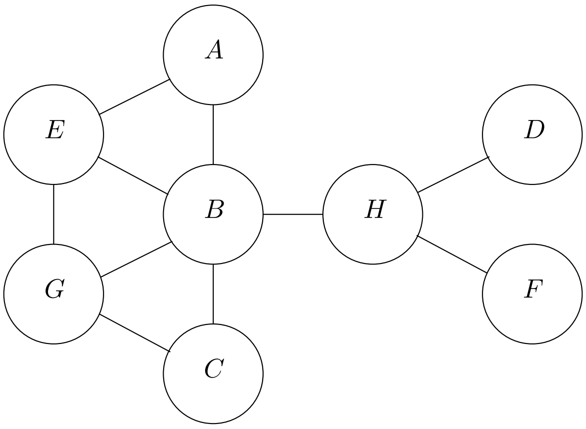
- ○ Yes
- ○ No

Solution:

A

2.2 CSP (5 pt)

Consider the following constraint satisfaction problem (CSP) with variables A, B, C, D, E, F, G, H and domains $\{R, S, T\}$. Each variable will be assigned a value R, S or T. The following graph shows the constraints between variables. Adjacent variables must have different value assignments.



2.2.1 Backtracking (2 pt)

Suppose we visit the variables in the order A, B, C, D, E, F, G, H and assign the values in the order R, S, T. Run backtracking search on the given CSP. Let X be the variable that causes the first failure. Under different filtering strategies, which variable X will cause backtracking?

(a) **No filtering.** (1 pt)

○ C ○ E ○ G ○ No backtracking

(b) **Forward checking.** (0.5 pt)

○ C ○ E ○ G ○ No backtracking

(c) **Arc consistency.** (0.5 pt)

○ C ○ E ○ G ○ No backtracking

Solution:

G, E, No backtracking

2.2.2 Minimum Remaining Values (1 pt)

Run backtracking search with the minimum remaining values (MRV) heuristic on the given CSP. Suppose we are doing forward checking and we have assigned $A \leftarrow R$, $B \leftarrow S$. Which variable will be assigned next?

○ C ○ D ○ E ○ F ○ G ○ H

Solution:

E

2.2.3 Least Constraining Value (1 pt)

Run backtracking search with the least constraining value (LCV) heuristic on the given CSP. Suppose we are enforcing arc-consistency and we have assigned $A \leftarrow R$, $B \leftarrow S, C \leftarrow T$. Next we are considering assigning D. Which value will be assigned to D?

○ R ○ S ○ T

Solution:

S

2.2.4 Cutset Conditioning (1 pt)

In cutset conditioning, we find a small set of variables (cutset) in the CSP to make the remaining part a tree-structured CSP. We then instantiate the cutset variables and solve the remaining CSP in polynomial time. Which set of variables is a minimum cutset for the given CSP? A minimum cutset is the cutset that has the minimum number of variables.

- ○ $\emptyset$
- ○ $\{B\}$
- ○ $\{B, H\}$
- ○ $\{B, H, D, F\}$
- ○ $\{A, B, C, D, E, F, G, H\}$

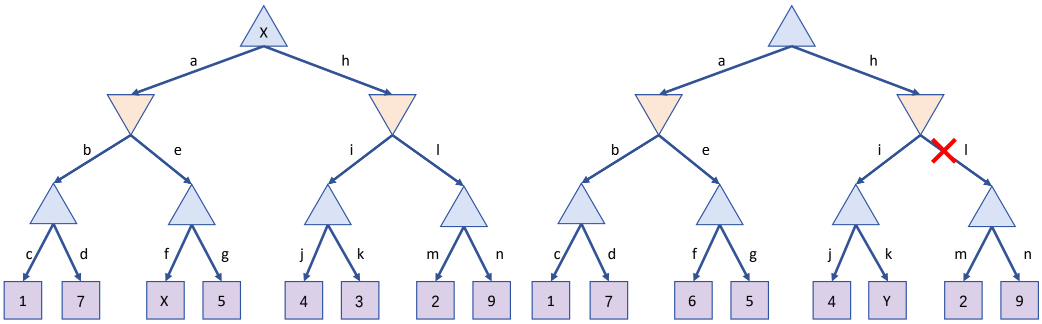
Solution:

$\{B\}$

3 Game (10 pt)

In all the sub-problems, the MAX nodes are represented by an upward triangle ($\triangle$), the MIN nodes are represented by a downward triangle ($\bigtriangledown$) and the chance nodes are represented by a circle.

3.1 Minimax Search (3 pt)



(a) Tree for question (1) (b) Tree for question (2)

Figure 2: Minimax search tree

Recall in the class we have the minimax tree as show in Figure 2. We have unknown value X in part (a) and Y in part (b) in the leaf nodes. The values for all nodes are real number ranging from $[0, 10]$. We evaluate the nodes from left to right. For your answers in the following 2 questions, you don't need to specify $X, Y \in [0, 10]$.

(1). We see part (a) in Figure 2. In this question, the values in all leaf nodes are distinct from each other. In order to have X as the value of the root node (i.e., we have to choose the "$a - e - f$" branch), what is the range of X? (1 pt)

$$\boxed{} < X < \boxed{}$$

(2). Now we see part (b) in Figure 2. In order to have l to be pruned, what is the range of Y? (2 pt)

$$0 \leq Y \leq \boxed{}$$

Solution:

(1) $5 < X < 7$

(2) $Y \leq 6$

3.2 Chance node (3 pt)

We use a circle to represent a chance node as shown in Figure 3, in which an agent makes its decision according to taking weighted average of children nodes. By default, we assume equal probabilities for each child node (e.g., if there are two child nodes, each is assigned a probability of 0.5). Assume that the leaf nodes are to be evaluated in a left-to-right order, and that before a leaf node is evaluated, we know nothing about its value. The values for all nodes are real number ranging from $[0, 10]$.

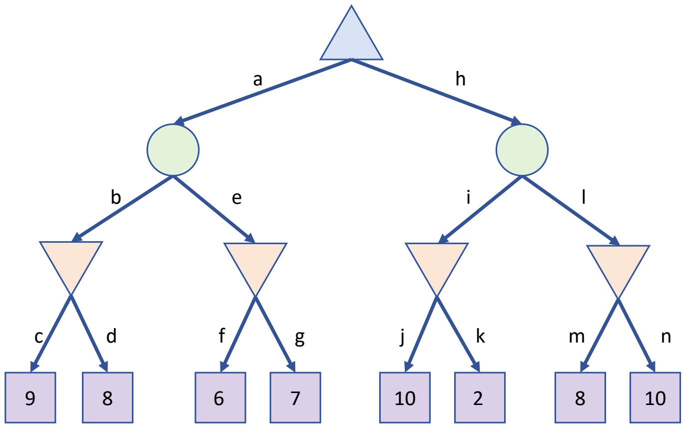


Figure 3: Game tree for Chance node

(1). Determine which branch(es) $(a, b, c, ..., l, m, n)$ can be pruned in the game tree in Figure 3? (1 pt)

☐ a ☐ b ☐ c ☐ d ☐ e ☐ f ☐ g

☐ h ☐ i ☐ j ☐ k ☐ l ☐ m ☐ n

☐ None

(2). Now we are going to change the distribution for the change nodes to be $(0.2, 0.8)$ and 0.2 for the left child node and 0.8 for the right child node, please get the final value for the root node. (1 pt)

(3). Can we do the exactly the same pruning as in (1) of the tree in (2)?(1 pt)

○ Yes. ○ No.

Solution:

(1) l or (l, m, n) ((m, n) is not correct but we give full points)

(2) 6.8

(3) No, we cannot prune in (2)

3.3 Multi-Agent Utilities (4 pt)

Consider a three-player non-zero-sum game with players A, B, and C (in Figure 4) and they are all maximizer. Each player wants to maximize their own utility, but they can also form alliances to potentially increase their combined utilities. At each decision point, the players can either act **independently** or form a **coalition** with one or more players to maximize their mutual benefit. Specifically, if player A and C form a coalition, when they decide their value, they prefer the larger total utility of A and C. For example, when player A and C need to determine the value, and they have 2 choices (a, b, c) and (d, e, f), if $a + c$ is larger than $d + f$ the former one will be chosen.

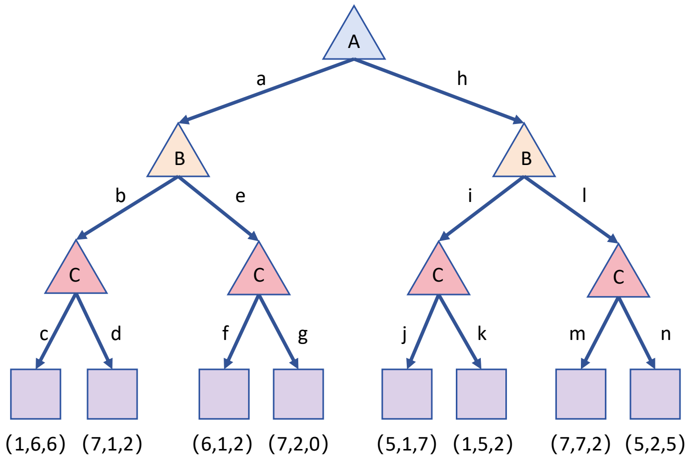


Figure 4: Game tree for a non-zero-sum game with 3 players.

(1). Please determining the value for the root node if all the players act independently. (Your answer should be a tuple with 3 values) (2 pt)

(2). Now assume we have a coalition between player B and C, please get the final value for the root node. (1 pt)

(3). Does forming a coalition always guarantee that the members will achieve a higher utility than they would if they acted independently? (1 pt)

○ Yes. ○ No.

Solution:

(1) (5,2,5)

(2) (7,7,2)

(3) No

4 Logic (10 pt)

4.1 Propositional logic (4 pt)

Leo is a kid from Argentina who likes to play football. His dream since childhood is to become the best football player in the world like Cristiano. He visited famous coaches everywhere until he met a great coach in Manchester and became his apprentice. The coach often gave Leo some useful suggestions. Leo had a bad memory, so he wrote down these suggestions in the form of propositional logic. But after a while, Leo could not connect the logical variables with the corresponding semantics. Can you help him? This is very important to help him become the best football player. It's worth noting that the order of Coach Suggestion on the left and Propositional Logic on the right does not completely correspond.

Coach Suggestion	**Propositional Logic**
If you drink milk and keep practicing, you will be strong.	$(\neg C \vee A) \wedge (\neg C \vee D)$
The best football player must be strong and keep practicing.	$\neg B \vee \neg D \vee A$
Keep practicing or drink milk every day, you can also do them both.	$\neg C$
You are not the best football player.	$B \vee D$

1. Drink milk:

   ○ A ○ B ○ C ○ D

2. Be strong:

   ○ A ○ B ○ C ○ D

3. Keep practicing:

   ○ A ○ B ○ C ○ D

4. Be the best football player:

   ○ A ○ B ○ C ○ D

Solution:

(1) B

(2) A

(3) D

(4) C

4.2 Conjunctive Normal Form (1 pt)

Leo was not satisfied with these suggestions, so he used these logical variables to randomly generate new propositional logic: $(A \wedge C \wedge D) \vee C \Rightarrow B$. To make it easier to understand, you can help him convert the new propositional logic into **Conjunctive Normal Form** (CNF).

Solution:

$(\neg A \vee B \vee \neg C \vee \neg D) \wedge (B \vee \neg C)$
or
$B \vee \neg C$

4.3 Resolution (2 pt)

Finally, we can use all the propositional logic mentioned above to construct a Knowledge Base (KB). Please judge if $KB \models \neg C \wedge \neg D$ is true, if it is true, use the resolution algorithm to prove your answer below.

Solution:

False
Convert $KB \wedge \neg(\neg C \wedge \neg D)$ to CNF:

$\neg C \wedge (B \vee D) \wedge (\neg B \vee \neg D \vee A) \wedge (\neg C \vee A) \wedge (\neg C \vee D) \wedge (\neg A \vee B \vee \neg C \vee \neg D) \wedge (B \vee \neg C) \wedge (C \vee D)$

Repeatedly apply the resolution rule to add new clauses. Until there is no new clause that can be added, empty set is not yielded. Therefore, KB does not entail $\neg C \wedge \neg D$.
A counterexample can also prove it:
When A=True, B=True, C=False, D=True, we have KB is True, but $\neg C \wedge \neg D$ is False

4.4 First order logic translation (3 pt)

Now we need to organize the information with first-order logic. First, we define several symbols.

- Footballplayer(x): x is a football player.
- Bestplayer(x, y): x is the best football player of y.
- in(x, y): x is in y.
- Goal(x): The goal number of x.
- $>(x, y)$: x is more than y.
- $=(x, y)$: x is equal to y.

Convert the following description into first-order logic clauses.

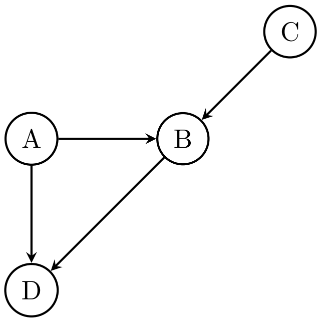
Description	**First-order Logic**
At least one football player has scored more than 100 goals.	
For any Chinese football player, there is always a French player who has scored the same number of goals.	
England's best football player has scored more goals than any other player in the country	

Solution:

Description	**First-order Logic**
At least one football player has scored more than 100 goals.	$\exists x \text{ Footballplayer}(x) \wedge > (Goal(x), 100)$
For any Chinese football player, there is always a French player who has scored the same number of goals.	$\forall x \text{ Footballplayer}(x) \wedge \text{in}(x, \text{Chinese}) \Rightarrow$ $\exists y \text{ Footballplayer}(y) \wedge \text{in}(y, \text{French}) \wedge = (\text{Goal}(x), \text{Goal}(y))$
England's best football player has scored more goals than any other player in the country	$\exists x \text{ Bestplayer}(x, \text{England}) \wedge (\forall y \text{ Footballplayer}(y) \wedge$ $\text{in}(y, \text{England}) \wedge \neg = (x, y) \Rightarrow > (\text{Goal}(x), \text{Goal}(y)))$

5 Graphic Models (10 pt)

Consider the Bayes net shown in the figure below.



Given the following conditional probability tables:

A	$P(A)$
$+a$	0.05
$-a$	0.95

C	$P(C)$
$+c$	0.3
$-c$	0.7

			$P(B\|A,C)$
$+a$	$+c$	$+b$	0.65
$+a$	$+c$	$-b$	0.35
$+a$	$-c$	$+b$	0.15
$+a$	$-c$	$-b$	0.85
$-a$	$+c$	$+b$	0.25
$-a$	$+c$	$-b$	0.75
$-a$	$-c$	$+b$	0.55
$-a$	$-c$	$-b$	0.45

			$P(D\|A,B)$
$+a$	$+b$	$+d$	0.60
$+a$	$+b$	$-d$	0.40
$+a$	$-b$	$+d$	0.10
$+a$	$-b$	$-d$	0.90
$-a$	$+b$	$+d$	0.20
$-a$	$+b$	$-d$	0.80
$-a$	$-b$	$+d$	0.50
$-a$	$-b$	$-d$	0.50

5.1 Exact Inference (5 pt)

5.1.1 Markov blanket (1 pt)

Choose all the variables that belong to the Markov blanket of variable A.

☐ B ☐ C ☐ D

Solution:

B C D

5.1.2 Probabilities (4 pt)

Which of the following are asserted by the network structure? Choose "yes" if the statement is asserted by the network structure, and "no" otherwise.

1. $D \perp\!\!\!\perp C \mid A, B$ ○ yes ○ no

2. $A \perp\!\!\!\perp C \mid B, D$ ○ yes ○ no

3. $A \perp\!\!\!\perp C$ ○ yes ○ no

Solution:

yes no yes

Using the Bayes' Net and conditional probability tables above, calculate the following quantity.

$P(+a, +b, +d) =$ ☐

Solution:

0.009

5.2 Approximate Inference (5 pt)

5.2.1 Likelihood Weighting (3 pt)

Suppose that we want to use likelihood weighted sampling to approximate $P(A| + b, +c, +d)$. However, we accidentally forgot to fix the value of C and D, and instead we sampled them just like unconditioned variables!

For each of the samples below, write what the weight of the sample should be, in order to correctly approximate $P(A| + b, +c, +d)$. If the weight of the sample does not matter for calculating $P(A| + b, +c, +d)$, write 'reject' instead (since we would not use that sample).

$(+a, +b, -c, +d)$: ☐

$(+a, +b, +c, +d)$: ☐

$(-a, +b, +c, +d)$: ☐

Solution:

reject, 0.65, 0.25

5.2.2 Gibbs Sampling (2 pt)

Let's say we're trying to approximate $P(A| - b)$ using Gibbs sampling. Suppose the most recent sample is $(+a, -b, +c, +d)$ .If we choose D to resample, what is the probability of resampling $+d$ and $-d$ respectively?

$+d$: ☐

$-d$: $\boxed{}$

Solution:

0.10, 0.90