UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFORMATICS 2D: REASONING AND AGENTS

Thursday 16 August 2012

09:30 to 11:30

Convener: J Bradfield
External Examiner: A Preece

INSTRUCTIONS TO CANDIDATES

1. **Candidates in the third or later year of study for the degrees of MA(General), BA(Relig Stud), BD, BCom, BSc(Social Science), BSc (Science) and BEng should put a tick ($\surd$) in the box on the front cover of the script book.**
2. **Answer Parts A, B and C. The multiple choice questions in Part A are worth 50% in total and are each worth the same amount. Mark one answer only for each question - multiple answers will score 0. Marks will not be deducted for incorrect multiple choice exam answers. Raw multiple choice scores may be rescaled at the discretion of the exam board. Parts B and C are each worth 25%. Answer ONE question from Part B and ONE question from Part C.**
3. **Use the special mark sheet for Part A. Answer Parts B and C each in a separate script book.**

Write as legibly as possible.
CALCULATORS MAY BE USED IN THIS EXAMINATION.

Part A

ANSWER ALL QUESTIONS IN PART A. Use the special mark sheet.

1. Below we give 5 statements about the values assigned to the 4 propositional variables: P, B_1, B_2, and B_3. Which of these 5 statements describes *all* and *only* the assignments that make the following propositional formula true:

$$P \Leftrightarrow (B_1 \vee B_2 \vee B_3)$$

   (a) P is false and all of the B_i are false.

   (b) P is true and at least one of the B_i is true.

   (c) P is true and all of the B_i are true.

   (d) Either P is true and all of the B_i are true, or P is false and at least one of the B_i is false.

   (e) Either P is true and at least one of the B_i is true, or P is false and all of the B_i are false.

2. In a vocabulary with only 3 propositional symbols A, B, and C, how many models are there for the sentence $A \Rightarrow B$?

   (a) 3

   (b) 0

   (c) 1

   (d) 6

   (e) 2

3. Which of the following statements is *false* for binary resolution?

   (a) It is complete when combined with factoring.

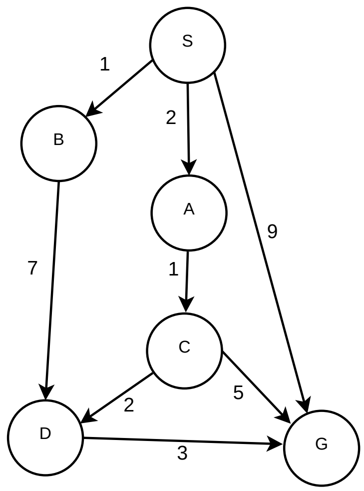
   (b) It may not always terminate.

   (c) It is restricted to binary clauses.

   (d) It can be used with both propositional and first order clauses.

   (e) It is only applicable to clauses that are in conjunctive normal form (CNF).

4. Consider the following graph, representing a search problem in which S is the start node and G is the only goal node. The value next to each directed edge $X \rightarrow Y$ represents the cost of going from node X to its successor Y. The heuristic function, which gives the estimated cost from any node to the goal, is given as a table. In what order will an A* GRAPH-SEARCH explore the graph?



S	A	B	C	D	G
1	0	2	1	5	0

(a) S, G
(b) S, A, B, D, G
(c) S, A, B, C, G
(d) S, A, C, G
(e) S, A, C, B, G

5. If b is maximum branching factor of a search tree, d is the depth of the least-cost solution and m is the maximum depth of the search space, what is the time complexity of α/β pruning, assuming a perfect ordering of the nodes?

(a) $O(b^{m/2})$
(b) $O(b^m)$
(c) $O(\sqrt{b})$
(d) $O(b^{\sqrt{d}})$
(e) $O(b^{d/m})$

6. Which of the following statements about backward chaining is *false*?

   (a) It is used in logic programming.
   (b) It cannot make redundant inferences.
   (c) It has linear space requirements when used with depth-first search.
   (d) It is not complele for definite clauses.
   (e) It can suffer from infinite loops.

7. Consider a constraint satisfaction problem (CSP) with variables A, B, C that can take values from the domain $\{1, 2, 3, 4\}$ and the following constraints only:

   - $A > B$
   - $B > C$

   Assuming that a partial assignment with $A = 4$ has been generated, and that the next variable to be chosen is B, what would the *least constraining value* heuristic do next?

   (a) Assign $B = 1$.
   (b) Assign $B = \{3, 4\}$.
   (c) Assign $B = 4$.
   (d) Delete B from the problem.
   (e) Assign $B = 3$.

8. In the context of the situation calculus, recall that fluents are functions and predicates that can vary from one situation to the next. Which one of the following statements best describes the form taken by a successor-state axiom?

   (a) Action is possible $\Leftrightarrow$ Action's effect made Fluent true

   (b) Action is possible $\Rightarrow$
   (Fluent is true in result state $\Leftrightarrow$

   $$\begin{aligned}&\text{(It was true before and action left it alone}\\ &\qquad\vee \text{ Action's effect made it true))}\end{aligned}$$

   (c) Action is possible $\Rightarrow$

   $$\begin{aligned}&\text{(Fluent was true before and action left it alone}\\ &\qquad\vee \text{ Action's effect made it true)}\end{aligned}$$

   (d) Action is possible $\Rightarrow$
   (Fluent is true in result state $\Leftrightarrow$ Action's effect made it true)

   (e) Action is possible $\Rightarrow$
   (Fluent is true in result state $\Rightarrow$

   $$\begin{aligned}&\text{(Fluent was true before and action left it alone}\\ &\qquad\wedge \text{ Action's effect made it true))}\end{aligned}$$

9. Which of the following clauses is the result of resolving clause $\neg P(x, F(x))$ with clause $P(F(y), v) \vee Q(y, F(v))$?

   (a) $Q(y, F(F(x)))$

   (b) Resolution fails.

   (c) $P(F(y), F(F(y))) \vee Q(y, F(F(F(y))))$

   (d) $Q(y, F(v))$

   (e) $Q(y, F(F(F(y))))$

10. Let max_flips be the maximum number of times the values of variables can be randomly flipped and p be the probability of choosing to do a *random walk* in WALKSAT (as given in Russell & Norvig and the lectures). Which one of the following statements is *false*?

   (a) If max_flips is set to infinity and $p > 0$ then the algorithm will always terminate if the sentence is satisfiable.

   (b) If $p > 0$ and the algorithm returns failure then either the sentence is unsatisfiable or the value of max_flips is too low.

   (c) If $p = 1$ then the algorithm randomly picks a symbol from the current clause and flips its value.

   (d) If $p = 1$ then the algorithm randomly selects a clause that is false in the current model.

   (e) If $p = 0$ then the algorithm flips whichever symbol in the current clause which maximises the number of satisfied clauses.

11. "Contingency planning that includes sensing actions and a description of different paths for different circumstances" is an accurate description for which of the following planning techniques?

    (a) Replanning and execution monitoring

    (b) Conditional planning

    (c) Sensorless/Conformant planning

    (d) Hierarchical task network planning

    (e) Continuous planning

12. Using action schema

$$\begin{aligned}&\mathit{Action}(Walk(from, to),\\&\quad \textsc{Precond}{:}\mathit{At}(from), City(to), from \neq to\\&\quad \textsc{Effect}{:}(\mathit{At}(to), \neg \mathit{At}(from))\end{aligned}$$

    which of the following is the correct set of states that result from applying all possible actions in state $\{\mathit{At}(\mathit{Manchester}), \mathit{City}(\mathit{Glasgow}), \mathit{At}(\mathit{Home})\}$?

    (a) $\{\{\mathit{At}(\mathit{Manchester}), \mathit{City}(\mathit{Glasgow})\},$
    $\{\mathit{At}(\mathit{Glasgow}), \mathit{City}(\mathit{Glasgow})\}\}$

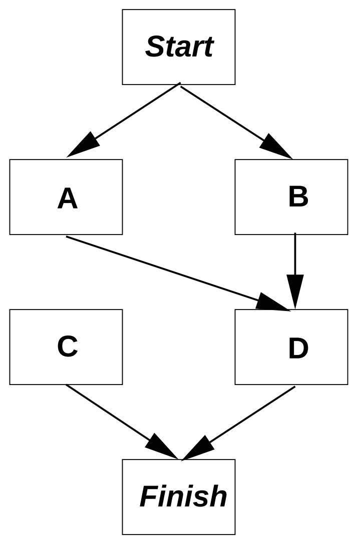
    (b) $\{\{\mathit{At}(\mathit{Glasgow}), \mathit{At}(\mathit{Home}), \mathit{City}(\mathit{Glasgow})\},$
    $\{\mathit{At}(\mathit{Glasgow}), \mathit{At}(\mathit{Manchester}), \mathit{City}(\mathit{Glasgow})\}\}$

    (c) $\{\{\mathit{At}(\mathit{Glasgow}), \mathit{City}(\mathit{Glasgow})\}\}$

    (d) $\{\{\mathit{At}(\mathit{Glasgow}), \mathit{City}(\mathit{Glasgow})\},$
    $\{\mathit{At}(\mathit{Glasgow}), \mathit{At}(\mathit{Home}), \mathit{At}(\mathit{Manchester}), \mathit{City}(\mathit{Glasgow})\}\}$

    (e) $\{\{\mathit{At}(\mathit{Glasgow}), \mathit{At}(\mathit{Manchester}), \mathit{At}(\mathit{Home}), \mathit{City}(\mathit{Glasgow})\}\}$

13. The following diagram shows a partial-order plan consisting of four steps A, B, C, D, where an arrow $x \rightarrow y$ between two actions denotes that action x must be executed before action y (and all actions must come after *Start* and before *Finish*, of course):



How many linearisations exist for this plan?

(a) 1

(b) 4

(c) 6

(d) 8

(e) 12

14. Consider the following 'Blocks World' PDDL planning operator:

$$Action(Move(b, x, y),$$
$$\textsc{Precond}{:}On(b, x) \land Clear(b) \land Clear(y)$$
$$\textsc{Effect}{:}On(b, y) \land Clear(x) \land \neg On(b, x) \land \neg Clear(y))$$

and two states S_1 and S_2 defined as follows:

S_1 : $\{On(A, B), Clear(A), Clear(C)\}$

S_2 : $\{On(X, Y), Clear(X), On(Y, Z), Clear(D)\}$

Which of the following statements is incorrect?

(a) $Move(A, B, C)$ can be applied in S_1

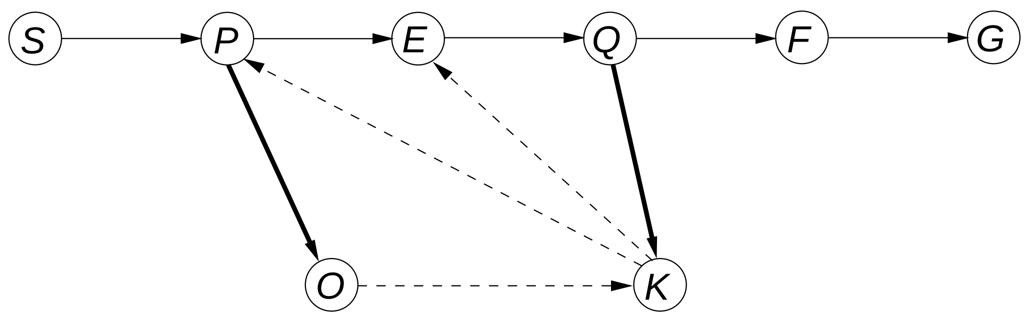
(b) Applying $Move(A, B, C)$ in S_1 would result in state $\{On(A, C), Clear(C), Clear(B)\}$

(c) $Move(X, Y, Z)$ cannot be applied in S_2

(d) Applying $Move(X, Y, D)$ in S_2 would result in state $\{On(X, D), Clear(X), On(Y, Z), Clear(Y)\}$

(e) $Move(X, Y, D)$ can be applied in S_2

15. The following diagram shows possible reactions to failure using replanning, where the an original (linear) plan goes through the sequence of states $\langle S, P, E, Q, F, G \rangle$:



Assuming that the cost of different sub-plans is proportional to the geometric distances between the states, what is the optimal replanning procedure the agent should execute if it ends up in unexpected states O and/or K (dashed lines denote possible intermediate goals during replanning, bold lines denote the only possible failures in this domain)?

(a) when in O, plan for E; when in K, plan for E

(b) when in O, plan for K; when in K, plan for P

(c) when in O, plan for P; when in K, plan for E

(d) when in O, plan for P; when in K, plan for P

(e) when in O, plan for E; when in K, plan for P

16. The process of computing $\vec{P}(\vec{X}_{t+k}|\vec{e}_{1:t})$ for $k > 0$ in a temporal probabilistic model with state variables $\vec{X}$ and evidence variables $\vec{e}$ is called

(a) smoothing

(b) abstraction

(c) monitoring

(d) prediction

(e) filtering

17. Assume the following inhibition probabilities between Boolean cause variables A, B, C and Boolean effect variable X:

$$P(\neg x|a, \neg b, \neg c) = p$$
$$P(\neg x|\neg a, b, \neg c) = q$$
$$P(\neg x|\neg a, \neg b, c) = r$$

What is the probability $P(x|a, b, \neg c)$ assuming that the conditional probabilities of X are computed using a noisy-OR relation?

(a) $1 - (p \times q)$

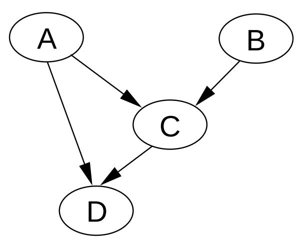
(b) $r \times p \times (1 - q)$

(c) $(1 - p) \times q \times r$

(d) $q \times (1 - p)$

(e) $(1 - r) \times (1 - p) \times q$

18. Consider the following Bayesian network structure with Boolean variables:



Which of the following statements is correct?

(a) All conditional probability tables in it have the same size.

(b) Given A, C is conditionally independent of B.

(c) Calculating the value of $P(a \wedge \neg b \wedge c \wedge \neg d)$ in this network involves using only multiplication of probabilities.

(d) All variables in the system are conditionally independent.

(e) It takes a total of $2^4 = 16$ probability values to describe the joint probability distribution represented by the network.

19. Assume two random variables X and Y are conditionally independent given a third variable Z. Which of the following statements is incorrect?

    (a) This assumption makes a reduced-size representation of the JPD possible

    (b) $\mathbf{P}(X, Y|Z) = \mathbf{P}(X|Z)\mathbf{P}(Y|Z)$

    (c) $\mathbf{P}(X|Y, Z) = \mathbf{P}(X|Z)$

    (d) $\mathbf{P}(Y|X, Z) = \mathbf{P}(Y|Z)$

    (e) $\mathbf{P}(X|Y) = \mathbf{P}(Y)$

20. Let strict preference be denoted by the relation symbol $\succ$, indifference by $\sim$, weak preference by $\succsim$, and lotteries be written as $[p_1, O_1; \ldots; p_n, O_n]$, where each p_i is the probability associated with outcome O_i. Which of the following is not an axiom of utility theory that must hold for any outcomes A, B, C?

    (a) $A \sim B \Rightarrow [p, A; 1-p, C] \succ [p, B; 1-p, C]$

    (b) $(A \succ B) \wedge (B \succ C) \Rightarrow (A \succ C)$

    (c) $(A \succ B) \vee (B \succ A) \vee (A \sim B)$

    (d) $A \succ B \Rightarrow (p \geq q \Leftrightarrow [p, A; 1-p, B] \succsim [q, A; 1-q, B])$

    (e) $A \succ B \succ C \Rightarrow \exists p\, [p, A; 1-p, C] \sim B$

Part B
ANSWER ONE QUESTION FROM PART B

1. Consider the following English sentences:

   > If the wumpus is invisible then it is undefeatable, but if it is not invisible then it is a defeatable monster. If the wumpus is either undefeatable or a monster then it is dangerous. The wumpus is a threat if it is dangerous.

   (a) Express the sentences above in *propositional logic*. Define all symbols used.

   [4%]

   (b) State the binary resolution rule of inference for propositional logic.

   [1%]

   (c) Convert your propositional logic sentences from part (1a) into clauses. Show all your work.

   [3%]

   (d) Using the clauses from part (1c), use binary resolution to prove that the wumpus is dangerous. Show all the steps of your proof.

   [5%]

   (e) State *and* describe the three improvements that the Davis-Putnam-Logemann-Loveland (DPLL) algorithm has over truth-table enumeration. Illustrate each of these improvements with an appropriate example.

   [12%]

2. Consider the following general constraint satisfaction problem in propositional logic involving n symbols:

$$(\neg X_1 \vee X_2) \wedge (\neg X_2 \vee X_3) \wedge \ldots \wedge (\neg X_{n-1} \vee X_n) \tag{1}$$

(a) Give a constraint graph for the CSP above when $n = 4$.

[3%]

(b) Briefly explain what the *minimum remaining value* heuristic is.

[4%]

(c) Briefly explain what the *least constraining value* heuristic is.

[4%]

(d) Briefly explain what is meant by constraint propagation?

[2%]

(e) For the given CSP, list the possible solutions when $n = 4$. How many solutions are there for a general problem with n symbols?

[5%]

(f) In the given CSP with $n = 4$, assume that the first assignment is $X_2 =$ True, give the domain for each of the variables after forward-checking. Explain your answer.

[3%]

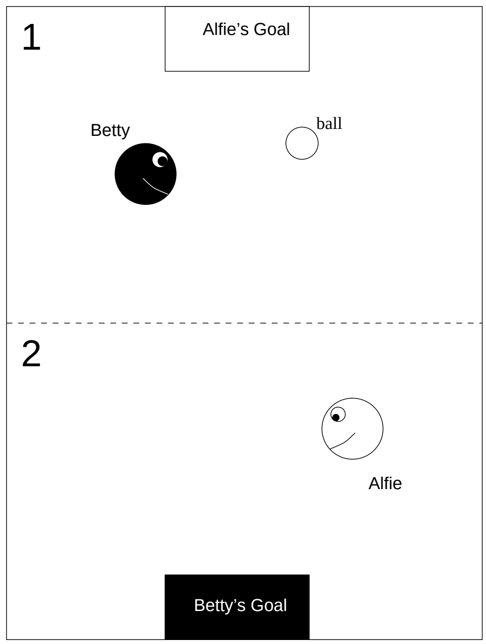
(g) In the given CSP with $n = 4$, assuming that you now start with $X_1 =$ True, give the domain for each of the variables after forward-checking *and* arc-consistency have been enforced. Explain your answer.

[4%]

Part C
ANSWER ONE QUESTION FROM PART C

1. **Planning**

   Consider the following agents playing a football (soccer) game:



You may assume that the agents can move into the regions 1 and 2, that they can shoot at the two goals, and they can take possession of the ball if the other player hasn't got it. The agents cannot distinguish between more specific positions than regions 1 and 2 and the goals. You are provided with the following domain language:

- *alfie* and *betty* (for the agents), *ball*, *r1* and *r2* (for regions 1 and 2), *goalA*, *goalB* (for Alfie and Betty's goal respectively).
- $Robot(a)$ denotes a is a robot; $Region(p)$ denotes that r is a region (true of *r1* and *r2*) and $Goal(g)$ denotes that g is a goal (true of *goalA* and *goalB*). $At(o, p)$ is true if the position of the ball or agent o is p. If o is *Betty* or *Alfie* then p can denote only $r1$ or $r2$ (i.e., the agents cannot be in *goalA* or *goalB*; if o denotes *ball* then p can denote $r1$, $r2$, *goalA* or *goalB*.
- *BallFree* denotes that no agent is in possession of the ball, and $HasBall(x)$ denotes that agent x has the ball.

(a) Define in PDDL the action schemata for the following actions:

i. *TakeBall*: allows an agent a to take possession of the ball, so long as a is in the same region as the ball and no agent is currently in possession of it. The effect of this action is that a has the ball.

[3%]

ii. *Move*: Allows an agent a to move from one region r' to another region r (where r and r' may denote the same region). If the agent moves while in possession of the ball, then the ball ends up in region r. However, a may lose possession of the ball.

[5%]

iii. *Shoot*: Allows an agent a to shoot at goal g, so long as a is in possession of the ball. If a is in the region in which g is in (e.g., r is *r1* and g is *goalB*), then r will score a goal (in other words, the action has the effect $At(ball, g)$). If, on the other hand, a shoots at g when in the other region (e.g., r is *r2* and g is *goalB*), then sometimes a successfully scores but sometimes the ball stays where it is with a losing possession of it.

[5%]

(b) Write in PDDL the initial state as shown in the above diagram.

[2%]

(c) Write in PDDL the goal state that the ball is in Betty's goal.

[1%]

(d) Write a plan in PDDL for Betty to achieve the goal state from the current state *without* moving to another region.
Hint: To handle steps that contain loops, you can use the syntax **while** *S* **do** *A, where S is a description of a state and A is an action. You may use [] to stipulate no action.*

[6%]

(e) Is this plan guaranteed to work? Explain your answer.

[3%]

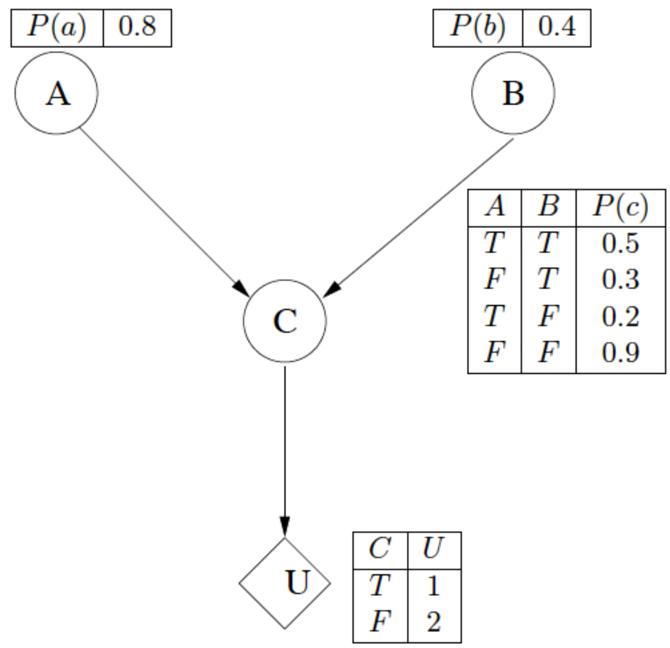


Figure 1: A Decision Network consisting of Boolean Chance Nodes

2. **Decision Networks**

The Decision Network in Figure 1 consists of three boolean chance nodes A, B and C with dependencies among them as shown, and a utility node U that's dependent on the value of C.

Note: Answers to all the following questions should be given to 3 decimal places.

(a) Calculate the joint probability $P(\neg a, b, \neg c)$ and the utility of this particular outcome.

[3%]

(b) Assume that we generate 100 samples using the above network to estimate the distribution $\mathbf{P}(B|c)$ and obtain the following results:

	b	$\neg b$
c	22	26
$\neg c$	24	28

What is the estimated distribution $\langle P(b|c), P(\neg b|c) \rangle$ that we would obtain by applying the rejection sampling method to this problem?

[3%]

(c) What is the actual probability distribution $\mathbf{P}(B|c)$ given by the conditional probability tables in Figure 1?

[9%]

(d) What is the expected utility that can be derived from the network?
Hint: Express this using sums of conditional probabilities and utilities, ensuring that each conditional probability in your formula is expressed directly in the conditional probability tables in the network.

[10%]

Specimen Answers

Part A

1. e
2. d
3. c
4. c
5. a
6. b
7. e
8. b
9. e
10. d
11. b
12. c
13. d
14. b
15. a
16. d
17. a
18. c
19. e
20. a

Part B

1. (a) Assuming we have propositional symbols defined as follows:

   - A: the wumpus is invisible
   - B: the wumpus is a monster
   - C: the wumpus is undefeatable
   - D: the wumpus is dangerous
   - E: the wumpus is a threat

   then we get the following propositional logic statements:

$$\begin{aligned} A &\Rightarrow C && (2) \\ \neg A &\Rightarrow \neg C \wedge B && (3) \\ B \vee C &\Rightarrow D && (4) \\ D &\Rightarrow E && (5) \end{aligned}$$

   Marking guide: 1 mark per sentence. Deduct 1 mark if the symbols are not defined.

   (b) (Bookwork) Binary resolution for propositional logic:

$$\frac{C \vee P \qquad D \vee \neg P}{C \vee D}$$

   (c) The following clauses result:

$$\begin{aligned} \neg A &\vee C && (6) \\ A &\vee \neg C && (7) \\ A &\vee B && (8) \\ \neg B &\vee D && (9) \\ \neg C &\vee D && (10) \\ \neg D &\vee E && (11) \end{aligned}$$

   Marking guide: Half a mark per clause.

(d) One possible proof: We negate the goal first to obtain

$$\neg D \tag{12}$$

Next:

- Resolve (6) with (8):

$$C \vee B \tag{13}$$

- Resolve (13) with (9):

$$C \vee D \tag{14}$$

- Resolve (14) with (10) (to give $D \vee D$) and then use the factoring rule:

$$D \tag{15}$$

- Resolve (15) with the negated goal (12) to give the empty clause:

$$\square \tag{16}$$

Marking guide: 1 mark for negating the goal and 4 marks for a successful proof. 2 marks for any reasonable attempt even if the empty clause is not derived. Deduct 1 mark if factoring is used but not mentioned explicitly.

(e) (Bookwork: Descriptions) The three improvements (as described in Russell & Norvig) are as follows:

i. *Early termination*: The algorithm detects whether the sentence must be true or false, even with a partially completed model. A clause is true is *any* of its literal is true, even if other literals have not been assigned any truth values; hence, the sentence as a whole could be judged true even before the model is complete. For example, the sentence $(A \vee B) \wedge (A \vee C)$ is true if A is true, regardless of the values of B and C. Similarly, a sentence is false if *any* clause is false, which occurs when each of its literals is false. Again, this can occur before the model is complete. Early termination thus avoids examination of entire subtrees in the search space.

ii. *Pure symbol heuristic*:
A pure symbol is one that always appears with the same sign (polarity) in all clauses. For instance, in the three clauses $\{A \vee \neg B, \neg B \vee \neg C, C \vee A\}$, the symbol A is pure because only the positive literals appear, B is pure because only the negative literals appear, and C is impure. If a sentence has a model, then it has one with the pure symbols assigned so as to make their literals *true*, because this ensures that a clause can never be false. In determining the purity of a symbol, the algorithm can ignore clauses

that are already known to be true in the model constructed so far. For instance, in our example, if the model contains $B = false$, then the clause $\neg B \vee \neg C$ is already true, and C then becomes pure because it now only appears in $C \vee A$.

iii. *Unit clause heuristic* A unit clause is one with just one literal. In the context of DPLL, it also means clauses in which all literals but one are already assigned *false* in the model. For example, if the model contains $B = false$ then $B \vee \neg C$ becomes a unit clause since it is equivalent to $false \vee \neg C$ i.e. $\neg C$. For this last clause to be true, C must be set to *false*. The unit clause heuristic assigns all such symbols before branching on the remainder.

Marking guide: 1 mark in each case for stating the name of the improvements (for a total of 3 marks). 2 marks for each concise and clear description – including definitions, wherever appropriate – of the improvements. 1 mark given for each of the three examples.

2. (a) The constraint graph with the propositions as nodes and constraints indicated by –: $X_1 - X_2 - X_3 - X_4$.

(b) (Bookwork): This heuristic involves choosing the variable with the fewest domain values. By picking the variable that is most likely to cause failure soon, the search tree can be pruned early. If there are no legal values left for a variable X, the heuristic will select X and failure will be detected immediately, forcing backtracking.

Marking guide: 4 marks for a clear and concise answer.

(c) (Bookwork) Once the variable has been chosen, this heuristics picks a value for it that constrains the other variables the least i.e. it prefers the value that rules out the fewest choices for the neighbouring variables in the constraint graph. Using this heuristic generally maintains maximal flexibility for subsequent variables assignment.

Marking guide: 4 marks for a clear and concise explanation.

(d) (Bookwork) Constraint propagation is a means of enforcing the implications of a constraint on one variable onto the other variables by looking ahead for inconsistencies.

Marking guide: 2 marks for explaining concisely what constraint propagation is.

(e) For $n = 4$, we have the following solutions:

X_1	X_2	X_3	X_4
False	False	False	False
False	False	False	True
False	False	True	True
False	True	True	True
True	True	True	True

In general, for n propositional symbols, we have $n+1$ solutions.

Marking guide: 3 marks for the solutions and 2 marks for the general formula.

(f) The domains after X_2=True but before forward-checking:
X_1: [True, False], X_2: [True], X_3: [True, False], X_4: [True, False]
After FC: X_1: [True, False], X_2: [True], X_3: [True], X_4: [True, False]

False is deleted from the domain of its neighbour X_3 because it is incompatible and would make the constraint $\neg X_2 \vee X_3$ false. The domain of its other neighbour, X_1, is unaffected.

Marking Guide: 2 marks for giving the correct domains. 2 marks for explanation.

(g) The domains after X_1=True but before arc-consistency:
X_1: [True], X_2: [True, False], X_3: [True, False], X_4: [True, False]
After arc-consistency: X_1: [True], X_2: [True], X_3: [True], X_4: [True]

With X_1=True, False is deleted from the domain of X_2 to ensure that the constraint $\neg X_1 \vee X_2$ is still satisfied. Additionally, though, False is eliminated from the domain of X_3 to ensure that $\neg X_2 \vee X_3$ is not violated and, as a result, False is also deleted from the domain of X_4 to ensure that the final constraint $\neg X_3 \vee X_4$ is also satisfied.

Marking Guide: 2 marks for giving the correct domains. 2 marks for explanation.

Part C

1. Planning

   (a) i.

   $$\begin{aligned}&\mathit{Action}(\mathit{TakeBall}(a),\\&\quad \textsc{Precond}{:}\mathit{Robot}(a) \wedge \mathit{At}(a,p) \wedge \mathit{At}(\mathit{ball},p) \wedge \mathit{FreeBall}\\&\quad \textsc{Effect}{:}\mathit{HasBall}(a))\end{aligned}$$

   ii.

   $$\begin{aligned}&\mathit{Action}(\mathit{Move}(a,r',r),\\&\quad \textsc{Precond}{:}\mathit{Robot}(a) \wedge \mathit{Region}(r) \wedge \mathit{Region}(r') \wedge \mathit{At}(a,r')\\&\quad \textsc{Effect}{:}\mathit{At}(a,r) \wedge \neg \mathit{At}(a,r') \wedge\\&\qquad\qquad \textbf{if } \mathit{HasBall}(a)\\&\qquad\qquad\qquad \textbf{then } \neg \mathit{At}(\mathit{ball},r') \wedge \mathit{At}(\mathit{ball},r) \wedge (\mathit{HasBall}(a) \vee \mathit{FreeBall}))\end{aligned}$$

   iii.

   $$\begin{aligned}&\mathit{Action}(\mathit{Shoot}(a,g),\\&\quad \textsc{Precond}{:}\mathit{Robot}(a) \wedge \mathit{Goal}(g) \wedge \mathit{HasBall}(a)\\&\quad \textsc{Effect}{:}\neg \mathit{HasBall}(a) \wedge \mathit{FreeBall} \wedge\\&\qquad\qquad \textbf{if } ((\mathit{At}(a,r1) \wedge g = \mathit{goalA}) \vee (\mathit{At}(a,r2) \wedge g = \mathit{goalB}))\\&\qquad\qquad\qquad \textbf{then } \mathit{At}(\mathit{ball},g)\\&\qquad\qquad\qquad \textbf{else } (\mathrm{At}(\mathit{ball},g) \vee (\mathit{At}(a,r) \wedge \mathit{At}(\mathit{ball},r))))\end{aligned}$$

   (b)

   $$\begin{aligned}&\mathit{Robot}(\mathit{alfie}) \wedge \mathit{Robot}(\mathit{betty}) \wedge\\&\mathit{FreeBall} \wedge \mathit{Region}(r1) \wedge \mathit{Region}(r2) \wedge \mathit{Goal}(\mathit{goalA}) \wedge \mathit{Goal}(\mathit{goalB}) \wedge\\&\mathit{At}(\mathit{ball},r1) \wedge \mathit{At}(\mathit{betty},r1) \wedge \mathit{At}(\mathit{alfie},r2)\end{aligned}$$

   Deduct 1 point for not including all non-fluents. Deduct another point if fluents are wrong or missing.

   (c) $\mathit{At}(\mathit{ball},\mathit{goalB})$

   0 points if this is wrong.

   (d)

   $$\begin{aligned}[\quad &\textbf{while } \neg \mathit{At}(\mathit{ball},\mathit{goalB})\\&\quad \textbf{do } \mathit{TakeBall}, \mathit{Shoot}(\mathit{betty},\mathit{goalB})]\end{aligned}$$

   Deduct 1 point if the plan includes the Move action. Deduct 2 point for missing the loop. Deduct 1 point if they put redundant steps in the plan.

(e) It's not guaranteed to work, because while Betty is trying to score, Alfie might move into region 1, and while the ball is not in possession of either he might take it and score in Alfie's goal. Once the ball is in *goalB*, Alfie cannot re-take possession of it and score.

Give 1 mark for saying plan can fail; and 2 marks for giving correct reason why.

2. Decision Networks

(a)

$$\begin{aligned} P(\neg a, b, \neg c) = &\ P(\neg a)P(b)P(\neg c|\neg a, b) \\ = &\ 0.2 \times 0.4 \times 0.3 \\ = &\ 0.024 \end{aligned}$$

$$U(\neg a, b, \neg c) = U(\neg c) = 2$$

One should drop 1 point for an error in the arithmetic, and 1 point for computing the wrong product of probabilities.

(b) There are 48 samples that satisfy c. Of these 22 are b and 26 or $\neg b$. Therefore, the estimated probability distribution is $\langle \frac{22}{48}, \frac{26}{48} \rangle = \langle 0.458, 0.542 \rangle$.

(c) The actual distribution is as follows:

$$\begin{aligned} \mathbf{P}(B|c) = &\ \textstyle\sum_a \frac{\mathbf{P}(a,B,c)}{P(c)} \\ \propto &\ \textstyle\sum_a P(a)\mathbf{P}(B)\mathbf{P}(c|B,a) \\ P(b|c) \propto &\ P(a)P(b)P(c|b,a) + P(\neg a)P(b)P(c|b,\neg a) \\ = &\ (0.8 \times 0.4 \times 0.5) + (0.2 \times 0.4 \times 0.3) \\ = &\ 0.16 + 0.024 \\ = &\ 0.184 \\ P(\neg b|c) \propto &\ P(a)P(\neg b)P(c|\neg b, a) + P(\neg a)P(\neg b)P(c|\neg a, \neg b) \\ = &\ (0.8 \times 0.6 \times 0.2) + (0.2 \times 0.6 \times 0.9) \\ = &\ 0.096 + 0.108 \\ = &\ 0.204 \\ \mathbf{P}(B|c) = &\ \tfrac{1}{(0.184+0.204)}\langle 0.184, 0.204 \rangle \\ = &\ \langle 0.474, 0.526 \rangle \end{aligned}$$

1 point should be deducted for each arithmetic error. 2 points should be deducted for each mistake in the Bayesian formulae or for calculations that are absent.

(d)

$$
\begin{aligned}
EU(N) = \;& P(c) + 2P(\neg c) \\
= \;& \textstyle\sum_a \sum_b (P(a,b,c) + 2P(a,b,\neg c)) \\
= \;& \textstyle\sum_a \sum_b P(a)P(b)(P(c|a,b) + 2P(\neg c|a,b)) \\
= \;& P(a)P(b)(P(c|a,b) + 2P(\neg c|a,b)) + \\
& P(\neg a)P(b)(P(c|\neg a,b) + 2P(\neg c|\neg a,b)) + \\
& P(a)P(\neg b)(P(c|a,\neg b) + 2P(\neg c|a,\neg b)) + \\
& P(\neg a)P(\neg b)(P(c|\neg a,\neg b) + 2P(\neg c|\neg a,\neg b)) \\
= \;& 0.8 * 0.4(0.5 + 2 * 0.5) + \\
& 0.2 * 0.4(0.3 + 2 * 0.7) + \\
& 0.8 * 0.6(0.2 + 2 * 0.8) + \\
& 0.2 * 0.6(0.9 + 2 * 0.1) \\
= \;& 0.48 + 0.136 + 0.864 + 0.132 \\
= \;& 1.612
\end{aligned}
$$

Deduct a point for each error.