Specimen Answers

Part A

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

Part B

1. (a) In the following exercise: $B(x)$ means "x is a baby", $L(x)$ means "x is logical", $D(x)$ means "x is despised" and $C(x)$ means "x can manage a crocodile". There are then four sentences:

$$\begin{aligned}
\forall x.B\ (x) &\rightarrow \neg L(x)\\
\forall x.\ C(x) &\rightarrow \neg D(x)\\
\forall x.\ \neg L(x) &\rightarrow D(x)\\
\forall x.\ B(x) &\rightarrow \neg C(x)
\end{aligned}$$

Marking guide: 1 mark for first sentence. 2 marks for the second one. 1 mark for the third sentence and remaining mark for the last sentence.

(b) (Bookwork) The binary resolution rule for FOL is defined as follows:

$$\frac{C \vee P \qquad D \vee \neg P'}{(C \vee D)\theta}$$

where θ is the most general unifier (MGU) of P and P'.

Marking guide: Deduct 1 mark if the MGU is not mentioned explicitly.

(c) For a proof by resolution the goal must first be negated:

Negated goal:

$$\begin{aligned}
&\neg\forall x.\ B(x) \rightarrow \neg C(x)\\
\equiv\ &\neg\forall x.\ \neg B(x) \vee \neg C(x)\\
\equiv\ &\exists x.\ B(x) \wedge C(x)
\end{aligned}$$

The following clauses thus result, after standardization apart of all variables:

- $\neg B(x) \vee \neg L(x)$
- $\neg C(y) \vee \neg D(y)$
- $L(z) \vee D(z)$
- $B(a)$ and $C(a)$, from the negated goal, where a is a Skolem constant.

One possible resolution proof: Resolve $B(a)$ with $\neg B(x) \vee \neg L(x)$ to yield $\neg L(a)$, which can in turn be resolved with $L(z) \vee D(z)$ to give $D(a)$. Resolve this against $\neg C(y) \vee \neg D(y)$ to yield $\neg C(a)$, which can then be resolved against the unit clause $C(a)$ to give the empty clause, thereby proving that the conclusion follows from the premises.

Marking guide: 1 mark for negating the conclusion. 4 marks for producing the clauses. 1 mark for clearly indicating that the negated goal produces a clause with a Skolem constant. 3 marks for the resolution proof.

(d) (Bookwork) The following algorithm is from Russell & Norvig:

function FOL-BC-ASK(KB, $query$) **returns** a generator of substitutions
  return FOL-BC-OR(KB, $query$, { })

---

generator FOL-BC-OR(KB, $goal$, θ) **yields** a substitution
  for each rule ($lhs \Rightarrow rhs$) in FETCH-RULES-FOR-GOAL(KB, $goal$) **do**
    (lhs, rhs) ← STANDARDIZE-VARIABLES((lhs, rhs))
    for each θ' **in** FOL-BC-AND(KB, lhs, UNIFY(rhs, $goal$, θ)) **do**
      yield θ'

---

generator FOL-BC-AND(KB, $goals$, θ) **yields** a substitution
  if θ = $failure$ **then return**
  else if LENGTH($goals$) = 0 **then yield** θ
  else do
    $first$,$rest$ ← FIRST($goals$), REST($goals$)
    for each θ' **in** FOL-BC-OR(KB, SUBST(θ, $first$), θ) **do**
      for each θ'' **in** FOL-BC-AND(KB, $rest$, θ') **do**
        yield θ''

Marking guide: The answer does not have to be exactly as above as long as a clear description of the algorithm is given.

2. This problem has some similarities to the Wumpus problem from Russell & Norvig and the lectures.

(a) The states can be characterized as sets of pairs of positions, where each position is itself a pair of natural numbers:

$$\{((x_1, y_1), (x_2, y_2)) | x_1, y_1, y_2, y_3 \in \{1, 2, \ldots, N\}\}$$

Marking guide: An answer describing the state rigorously in natural language is fine although a mathematical one is better.

(b) The maximum size of the state space is N^2 for both Romeo and Juliet. Thus, the maximum size for the problem is N^4.

(c) Romeo and Juliet have a choice of 5 actions each. Thus the maximum branching factor is $5^2 = 25$.

(d) The goal test simply involves checking for any given state $((x_1, y_1), (x_2, y_2))$ whether $x_1 = x_2 \wedge y_1 = y_2$.

(e) i. (Bookwork) A heuristic $h(n)$ is admissible if for every node n, $h(n) \leq h^*(n)$, where $h^*(n)$ is the true cost to reach goal state n. An admissible heuristic never over-estimates the cost to reach the goal i.e. it is optimistic.

ii. Let the Manhattan distance between Romeo and Juliet be d. Then a non-trivial admissible heuristic is $d/2$ since both of them take a step simultaneously.

(f) (Bookwork). The following is an acceptable answer:

i. Dominance (or domination): Heuristic h_2 dominates h_1 means that for any node n, $h_2(n) \geq h_1(n)$.

ii. (From R&N:) Domination translates directly into efficiency: A^* using h_2 will never expand more nodes than A^* using h_1, except possibly for some nodes with $f(n) = h^*(n)$. This is because every node with $f(n) < h^*(n)$ will surely be expanded. But because h_2 is at least as big as h_1 for all nodes, every node that is surely expanded by A^* using h_2 will also surely by A^* using h_1, and h_1 might cause further nodes to be expanded too.
Hence, it is generally better to use a heuristic function with higher values, provided it is consistent and that the computation time for the heuristic is not too long.

Marking Guide: For (2(f)ii), full marks can be given even if the last remark about computation time is omitted.

(g) Assume that n' is the successor of some node n generated by an action a and that $c(n, a, n')$ denotes the actual cost of this action. We proceed by induction: by the induction hypothesis (IH), $h(n') \leq h^*(n')$, where $h^*(x)$ denotes the true cost of reaching the goal for a node x. Now, we have:

$$\begin{aligned} h(n) &\leq c(n, a, n') + h(n') && \text{since } h \text{ is consistent} \\ &\leq c(n, a, n') + h^*(n') && \text{by IH} \\ &\leq h^*(n) \end{aligned}$$

and so h is admissible.

Marking Guide: This is a simple proof if induction is used (and the definition of consistency is recalled correctly). Give 2 marks for (use of) definition of consistency and 3 marks for the rest of the proof.

Part C

1. The Planning Question

   (a) Initial state is:

$$
\begin{aligned}
&horizontal(A) \wedge at(A,1,3) \wedge at(B,2,1) \wedge horizontal(B) \wedge \\
&vertical(C) \wedge at(C,3,2) \wedge \\
&occupy(1,1) \wedge occupy(1,3) \wedge occupy(2,1) \wedge occupy(2,3) \wedge \\
&clear(1,2) \wedge clear(2,2) \wedge clear(3,3)
\end{aligned}
$$

   The goal state is:

$$
\begin{aligned}
&at(A,2,3) \wedge at(C,3,1) \wedge horizontal(A) \wedge vertical(C) \wedge \\
&clear(1,3)
\end{aligned}
$$

   Marking guideline: Deduct 1 point if they use negations in the initial state: e.g., $\neg clear(1,2)$. Deduct 1 point for each missing literal. Deduct 1 point if the goal state is over specific compared with how it was described (i.e., vehicles A and C at the exit)—e.g., $clear(1,2)$.

   (b) i.

$$
\begin{aligned}
&Action(MoveRight(v,x,y),\\
&\quad \textsc{Precond:} at(v,x,y) \wedge horizontal(v) \wedge \\
&\qquad (((\neg clear(x+1,y) \wedge clear(x+2,y) \wedge (x+2 \leq 3)) \vee \\
&\qquad\qquad (\neg clear(x-1,y) \wedge clear(x+1,y) \wedge (x+1 \leq 3)))) \\
&\quad \textsc{Effect:} \neg at(v,x,y) \wedge at(v,x+1,y) \wedge \neg clear(x+1,y) \wedge \\
&\qquad (\textbf{when } \neg clear(x+1,y); \neg clear(x+2,y) \wedge clear(x,y)) \wedge \\
&\qquad (\textbf{when } \neg clear(x-1,y); \neg clear(x,y) \wedge clear(x-1,y)))
\end{aligned}
$$

   ii.

$$
\begin{aligned}
&Action(MoveLeft(v,x,y),\\
&\quad \textsc{Precond:} at(v,x,y) \wedge horizontal(v) \wedge \\
&\qquad (((\neg clear(x+1,y) \wedge clear(,x-1,y) \wedge (x-1 \geq 1)) \vee \\
&\qquad\qquad (\neg clear(x-1,y) \wedge clear(x-2,y) \wedge (x-2 \geq 1)))) \\
&\quad \textsc{Effect:} \neg at(v,x,y) \wedge at(v,x-1,y) \wedge \neg clear(x-1,y) \wedge \\
&\qquad (\textbf{when } \neg clear(x+1,y); clear(x,y) \wedge clear(x+1,y)) \wedge \\
&\qquad (\textbf{when } \neg clear(x-1,y); \neg clear(x-2,y) \wedge clear(x,y)))
\end{aligned}
$$

iii.

$$
\begin{aligned}
&\mathit{Action}(\mathit{MoveUp}(v,x,y),\\
&\quad \text{PRECOND:}\mathit{at}(v,x,y) \land \mathit{vertical}(v) \land\\
&\qquad (((\neg \mathit{clear}(x,y+1) \land \mathit{clear}(x,y+2) \land (y+2 \leq 3)) \lor\\
&\qquad\quad (\neg \mathit{clear}(x,y-1) \land \mathit{clear}(x,y+1) \land (y+1 \leq 3))))\\
&\quad \text{EFFECT:}\neg \mathit{at}(v,x,y) \land \mathit{at}(v,x,y+1) \land \neg \mathit{clear}(x,y+1) \land\\
&\qquad (\textbf{when } \neg \mathit{clear}(x,y+1); \neg \mathit{clear}(x,y+2) \land \mathit{clear}(x,y)) \land\\
&\qquad (\textbf{when } \neg \mathit{clear}(x,y-1); \mathit{clear}(x,y-1) \land \neg \mathit{clear}(x,y)))
\end{aligned}
$$

iv.

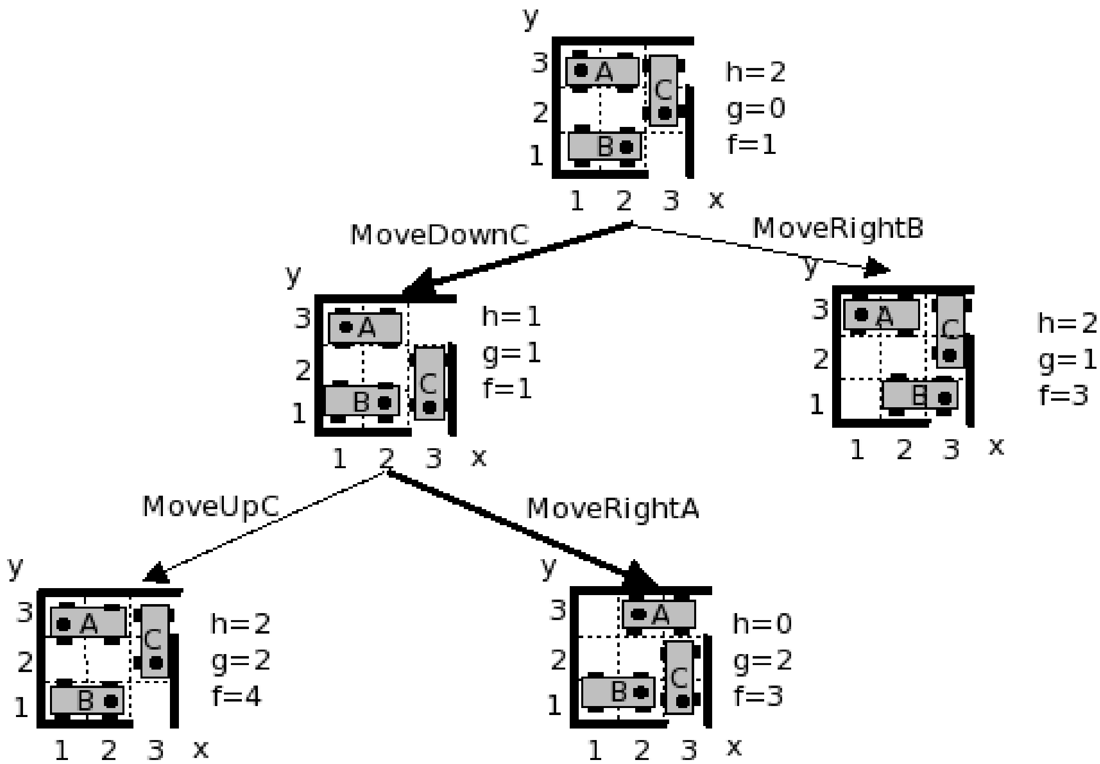
$$
\begin{aligned}
&\mathit{Action}(\mathit{MoveDown}(v,x,y),\\
&\quad \text{PRECOND:}\mathit{at}(v,x,y) \land \mathit{vertical}(v) \land\\
&\qquad (((\neg \mathit{clear}(x,y+1) \land \mathit{clear}(x,y-1) \land (y-1 \geq 1)) \lor\\
&\qquad\quad (\neg \mathit{clear}(x,y-1) \land \mathit{clear}(x,y-2) \land (y-2 \geq 1))))\\
&\quad \text{EFFECT:}\neg \mathit{at}(v,x,y) \land \mathit{at}(x,y-1) \land \neg \mathit{clear}(x,y-1) \land\\
&\qquad (\textbf{when } \neg \mathit{clear}(x,y+1); \neg \mathit{clear}(x,y) \land \mathit{clear}(x,y+1)) \land\\
&\qquad (\textbf{when } \neg \mathit{clear}(x,y-1); \neg \mathit{clear}(x,y-2) \land \mathit{clear}(x,y)))
\end{aligned}
$$

Marking guideline: Deduct 2 points if they decide to go beyond the vocabulary as its defined. Deduct 1 point for each missing literal (e.g., if they include $at(v, x, y)$ but not $\neg clear(x, y)$ in the effects), and 1 point for each unnecessary literal. If the same mistake is made in all 4 actions, then penalise only once.

(c) $h = 2$ (move A right, move C down or *vice versa*). $g = 0$ (because you haven't done any action in the initial state).

(d) h is consistent and admissible because it is a heuristic based on a relaxed constraint, and so it will always underestimate and never overestimate the true cost (in fact in fact for this planning problem the estimated cost of the initial state is the actual cost).

(e) The tree is as follows



Marking Guideline: Deduct 1 point if the costs are missing. Deduct 2 points for each missing state. Deduct 1 point if a state is in the tree that shouldn't be. Deduct 1 point if the optimal plan is wrong (or not shown). Deduct 1 point for each illegal move.

2. Bayesian Decision Network

(a)

$$\begin{aligned} P(\neg a, b, \neg c, d) = &\ P(\neg c|\neg ab)P(d|\neg ab)P(\neg a)P(b) \\ = &\ 0.5 * 0.6 * 0.2 * 0.4 \\ = &\ 0.024 \end{aligned}$$

(b)

$$\begin{aligned} \langle P(b|a, \neg d), P(\neg b|a, \neg d)\rangle = &\ \alpha\langle 2, 20\rangle \\ = &\ \langle 0.09, 0.91\rangle \end{aligned}$$

(c) The actual distribution is as follows:

$$\begin{aligned} \mathbf{P}(B|a, \neg d) = &\ \textstyle\sum_c \frac{\mathbf{P}(a,B,c,\neg d)}{P(a,\neg d)} \\ \propto &\ \textstyle\sum_c P(a)\mathbf{P}(B)\mathbf{P}(c|B, a)P(\neg d|B, a) \\ \propto &\ P(a)P(B)P(\neg d|B, a)\textstyle\sum_c \mathbf{P}(c|B, a) \\ \propto &\ P(a)P(B)P(\neg d|B, a) \\ P(b|a, \neg d) \propto &\ P(a)P(b)P(\neg d|b, a) = 0.8 * 0.4 * 0.1 = 0.032 \\ P(\neg b|a, \neg d) \propto &\ P(a)P(\neg b)P(\neg d|\neg b, a) = 0.8 * 0.6 * 0.4 = 0.192 \\ P(B|a, \neg d) = \langle 0.14, 0.86\rangle & \end{aligned}$$

(d)

$$\begin{aligned} EU(N) = &\ -1 * P(c, d) + 0 * P(\neg c, d) + 0 * P(c, \neg d) + 3 * P(\neg c, \neg d) \\ = &\ \textstyle\sum_a \sum_b (-1 * P(a, b, c, d) + 3 * P(a, b, \neg c, \neg d)) \\ = &\ -1 * \textstyle\sum_a \sum_b P(a, b, c, d) + 3 * \sum_a \sum_b P(a, b, \neg c, \neg d) \\ \textstyle\sum_a \sum_b P(a, b, c, d) = &\ P(a, b, c, d) + P(\neg a, b, c, d) + P(a, \neg b, c, d) + P(\neg a, \neg b, c, d) \\ = &\ P(a)P(b)P(c|a, b)P(d|a, b) + P(\neg a)P(b)P(c|\neg a, b)P(d|\neg a, b) + \\ &\qquad P(a)P(\neg b)P(c|a, \neg b)P(d|a, \neg b) + \\ &\qquad\quad P(\neg a)P(\neg b)P(c|\neg a, \neg b)P(d|\neg a, \neg b) \\ = &\ 0.8 * 0.4 * 0.2 * 0.9 + 0.2 * 0.4 * 0.5 * 0.6 + 0.8 * 0.6 * 0.7 * 0.6 + \\ &\qquad 0.2 * 0.6 * 0.9 * 0.2 \\ = &\ 0.0576 + 0.024 + 0.2016 + 0.0216 \\ = &\ 0.3048 \\ \textstyle\sum_a \sum_b P(a, b, \neg c, \neg d) = &\ P(a, b, \neg c, \neg d) + P(\neg a, b, \neg c, \neg d) + \\ &\qquad P(a, \neg b, \neg c, \neg d) + P(\neg a, \neg b, \neg c, \neg d) \\ = &\ 0.8 * 0.4 * 0.8 * 0.1 + 0.2 * 0.4 * 0.5 * 0.4 + \\ &\qquad 0.8 * 0.6 * 0.3 * 0.4 + 0.2 * 0.6 * 0.1 * 0.8 \\ = &\ 0.0256 + 0.016 + 0.0576 + 0.0096 \\ = &\ 0.1088 \\ EU(N) = &\ -0.3048 + (3 * 0.1088) \\ = &\ 0.0216 \end{aligned}$$