UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B. Sc. Examination 2015

Computing

IS53024A Artificial Intelligence

Duration: 2 hours and 15 minutes

Date and time:

There are five questions in this paper. You should answer no more than three questions. Full marks will be awarded for complete answers to a total of three questions. Each question carries 25 marks. The marks for each part of a question are indicated at the end of the part in [.] *brackets.*

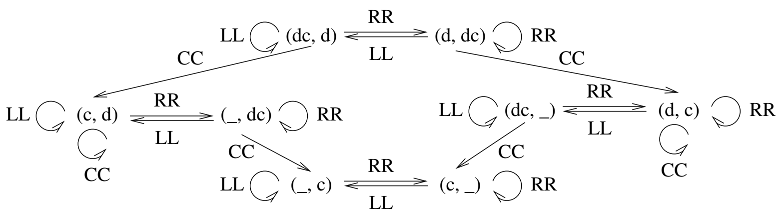
There are 75 marks available on this paper.

Electronic calculators must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.

THIS PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM

Question 1

(a) Consider the state transition diagram below for design of the *simple reflex agent* Whiteboard-Cleaner. One problem of the model is due to its rigid dependence on the environment states. Explain why this is a problem of the model. To demonstrate how the problem may be solved, redraw the diagram and highlight any changes. [9]



Assume the following notations:

i. Actions: `LL`: Move to the Left location; `RR`: Move to the Right location; `CC`: Clean the current location.

ii. States: $(\sqcup, \sqcup)$: the left and right locations; Each location can be clean "-" or dirty "d", and the cleaner "c" can be at either location. For example, (-, dc) depicts the state in which the left location is clean and the cleaner is at the right dirty location. Similarly, (c, d) depicts the state in which the cleaner is at the left clean location, and the right location is dirty. Note "-" is used only when the cleaner is absent from the location.

(b) Interpret the operation of a rule-based system considered to prove the goal (H) using the backward chaining algorithm. Assume the following rules and facts in the order in which they are given. [16]

```
(RULE 1 (IF ( E ) and ( A )) (THEN ( D )))
(RULE 2 (IF ( A ) and ( C )) (THEN ( D )))
(RULE 3 (IF ( D ) and ( F )) (THEN ( H )))
(RULE 4 (IF ( B ) and ( J )) (THEN ( F )))
(RULE 5 (IF ( G ) (THEN ( C )))
(FACT 1 ( G ))
(FACT 2 ( A ))
(FACT 3 ( B ))
(FACT 4 ( J ))
```

Question 2

Consider the task of stacking three boxes, [A], [B] and [C] vertically in a two-dimensional grid space so box A is on top of B, B is on top of C, and C is on the floor:

[A]
[B]
[C]

Initially, the three boxes are lined up horizontally on the floor as [A|B|C] so box A is on the left of box B and B is on the left of C. Assume that the agent to complete the task is capable of moving any *movable* box, one at a time, from its current location to a *free* location x. A box is *movable* if there is no box on top of it. A location is *free* if it is immediately on the floor or on top of a movable box.

Answer the questions below according to the description of the problem instance above.

(a) To help understand the problem, write down, as specifically as possible, all the legal moves from the intermediate situation ([A] on top of [C], beside [B]: [B|C]). [5]

(b) Design for the problem instance a state transition system in terms of *states* and *actions*. [7]

i. Draw a (partial) state-transition diagram to show how each action may cause *one* state transition from the initial state [A|B|C].

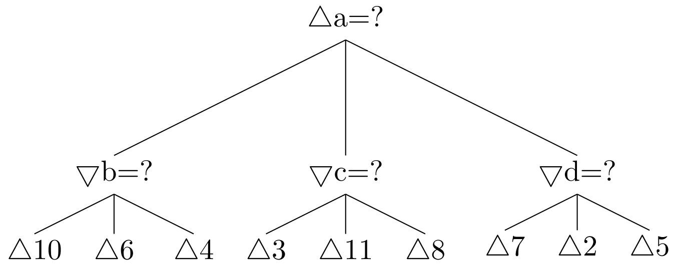
ii. Identify and highlight the optimal transition.

(c) Draw a series of diagrams to demonstrate how a search tree may be expanded step by step to reach the goal state applying the Depth-First Search (Graph-Search version) algorithm for the problem instance. Highlight the characteristics of the search algorithm and show all your work. [7]

(d) Demonstrate how the Greedy Best-First search algorithm may be applied for solving the problem instance. Trace step by step the expansion of the search tree, the evaluation and choice made on each state in execution of the algorithm. Highlight the characteristics of the algorithm and show all your work. [6]

Question 3

Consider the following tree for an adversarial game and answer each of the questions below.



(a) Assume the opponent is rational, and in the above figure the numbers at the leaf nodes are the values of the evaluation function applied. What are the backed-up values a, b, c and d that are computed using the MiniMax algorithm? Draw the complete game tree to show the values. Show all your work. [5]

(b) Demonstrate how the alpha-beta pruning algorithm works. Circle the nodes that are pruned using the alpha-beta algorithm. Assume child nodes are visited from the left to right. Show all your work. [10]

(c) Reorder the children of the root node a and the children of the three nodes b, c, d so that the alpha-beta algorithm would prune the maximum number of nodes possible. Assume children are visited from the left to right. Do not change any node's parent when you reorder. Again, circle the nodes that are pruned. [10]

Question 4

(a) Describe briefly how the candidate elimination algorithm for version space learning handles positive and negative examples. [6]

(b) Explain briefly when a generalisation is contained in a version space. [4]

(c) Consider a concept description language with 4 attributes given for symbolic machine learning. Suppose that these attributes can take the following values:

```
     a1           a2            a3           a4
 ---------      ------      ---------      ------
|    |    |    |      |    |    |    |    |      |
e    d    c    g      f    l    k    h    b      a
```

Demonstrate learning with the candidate elimination algorithm using the following positive and negative training examples:

```
1. ( c g l b ) +)
2. ( c f k a ) -)
3. ( e f l b ) +)
4. ( d g h b ) -)
5. ( d g l a ) -)
```

Illustrate the changes of the boundary sets after each example. [15]

Question 5

(a) Explain whether rule-based planning systems are inductive or deductive systems. [5]

(b) Give the four main elements of a rule-based planner. [4]

(c) Consider the problem of planning the transportation of air cargo. The planning involves the application of the forward chaining algorithm using the following specific rules with variables for air cargo planing:

```
(RULE 1  (IF   (plane ?p) (at ?p ?from) (airport ?from) (airport ?to))
         (THEN (ADD (at ?p ?to)
               (ADD (notat ?p ?from)))

(RULE 2  (IF   (plane ?p) (cargo ?c) (airport ?a) (at ?p ?a) (at ?c ?a) )
         (THEN (ADD (in ?c ?p))
               (ADD (notat ?c ?a)))

(RULE 3  (IF   (at ?p ?a) (in ?c ?p) (cargo ?c) (plane ?p) (airport ?a))
 (THEN (ADD (at ?c ?a))
               (ADD (notin ?c ?p)))
```

Assume that initially there is a given plane and cargo at a certain airport, that is the initial situation is defined with the following facts: (cargo C1) (plane P1) (airport SFO) (at C1 SFO) (at P1 SFO). Show the forward inference of the goal: (at C1 JFK) and give all variable bindings at each step of the algorithm. [16]