# 國立交通大學試題紙

九十七學年度第一次
博士班資格考

科目：人工智慧(A)　　日期：98 年 2 月 5 日　第 1 頁 共 3 頁

請"✓"明　✓不可看書　可看書

* 請將答案依題號順序寫入答案卷

答題時字跡需工整，否則不予計分。Write your answers legibly; otherwise you will get zero score.

1. Consider the eight-puzzle problem. There are various ways of representing the puzzle.
   (a) (4 pts) (Now one is to consider a move as the change from one board configuration to another. How many operators do we need?
   (b) (2 pts) Now if we consider a move as moving a blank in a given direction, then how many operators do we have?
2. (4 pts total, 1 pt each) Find the most general unifier (MGU) for the following pairs. Note x1, x2, x3 and x4 are variables. A, B and C are constants; f is a function.
   (a) a(x1, f(x2)) with a(A, B)
   (b) a(x1, x2, f(x1)) with a(g(x2), C, x3)
   (c) a(x1, B, f(x1,x2)) with a(A, x3, f(x3,x4))
   (d) a(x1, B, f(x1,x2)) with a(x3, x4, f(x3,x4))
3. (8 pts total) Given the following rules in predicate calculus. Prove (uses joe IBM-PC) by resolution.
   ∀ x (uses x IBM-PC) → (likes x segmented-architecture)
   ∀ x (desperate x) ∧ (has x IBM-PC) → (uses x IBM-PC)
   ∀ x (kicker x) → (desperate x)
   ∀ x (has x IBM-PC) → (likes x segmented-architecture)
   ∀ x (prefers x 8088-assembly) → (hacker x)
   ∀ x (prefers x 8088-assembly) → (likes x segmented-architecture)
   (prefers joe 8088-assembly)
   (has joe IBM-PC)
   Show all your work, including the conversion from predicate calculus to clause form.
4. (18 pts total) Our local shop has three machines, A, B and C, arranged on the floor in a particular linear order (e.g. A-B-C). We want to see if there is a linear arrangement of these machines which can produce a new part and meet a new time-efficiency requirement. Obviously, there are many permutations of machine orderings we can consider, but the catch is that due to the floor space limitation, we can only exchange any two machines at a time. Now consider this as a search problem. An operation is to exchange two machines.

(a) (2pts) How many search states do we need to check in the worst case? _____

(b) (2pts) What is the number of states to check in the best case? ____

Now we decide to use depth-first (DFS) or breadth-first search (BFS) to solve the problem, assuming DFS has cycle detection. Answer the following questions (True or False).

(c) (2 pts) DFS is STRICTLY better than BFS in the worst case? _____ (T or F)

(d) (2 pts) BFS and BFS give the same performance in the worst case? _____ (T or F)

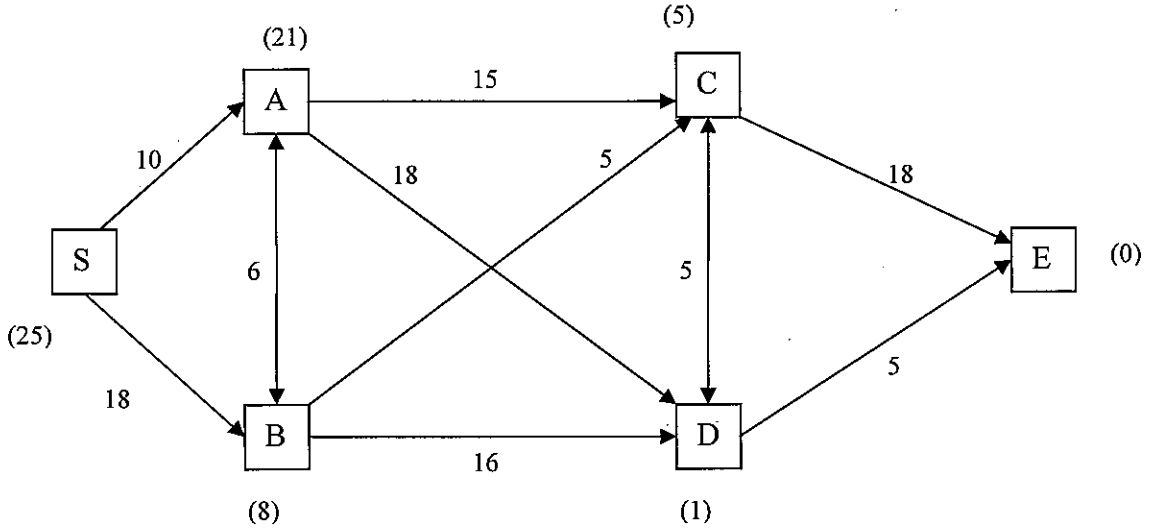
(e) (2 pts) DFS and BFS give the same performance in the best case? _____ (T or F)

Many engineers in the shop mention that configurations with A in the first position give best efficiency. Based on the HEURISTICS, what is the number of states to check in the worst case? (2 pts) ___

(f) (3 pts) Do you believe that if we exploit more background knowledge, we can get better average-case performance? Why or why not.

(g) (3 pts) Do you believe that if we exploit more background knowledge, we can get better worst-case performance? Why or why not.

5.




Given the search space above, where the numbers in parentheses are estimated cost and the numbers near the links are real cost, for each search strategy, write down the order in which nodes are expanded. Start from S and stop at E. Also assume that the only loop-checking being done is that a child which is also a direct parent of a node is never added to the queue, and there is no redundant-node check, i.e., expanded nodes are added to the queue anyway even if there is already the same node in the queue. If

two nodes or more in the queue have the same cost, they are taken off the queue in alphabetical order.

(a) (4 pts) Iterative-deepening DFS

(b) (3 pts) Depth-first search

(c) (3 pts) Hill-climbing

(d) (4 pts) A*

# 國 立 交 通 大 學 試 題 紙

九十七學年度第一次
博士班資格考

科目：人工智慧 (B)　　日期： 98 年 2 月 5 日　第 1 頁 共 1 頁

請" ✓" 明　　✓不可看書　　可看書

* 請將答案依題號順序寫入答案卷

答題時字跡需工整，否則不予計分。Write your answers legibly; otherwise you will get zero score.

1 Define in your own words the following terms:
(a) (3%) The frame problem;
(b) (3%) Learning agent.

2 (6%) Describe the differences and similarities between problem solving and planning.

3 The monkey-and-bananas problem is faced by a monkey in a laboratory with some bananas hanging out of reach from the ceiling. A box is available that will enable the monkey to reach the bananas if he climbs on it. Initially, the monkey is at *A*, the bananas at *B*, and the box at *C*. the monkey and box have height *Low*, but if the monkey climbs onto the box he will have height *High*, the same as the bananas. The actions available to the monkey include *Go* from one place to another, *Push* an object from one place to another, *ClimbUp* onto or *ClimbDown* from an object, and *Grasp* or *Ungrasp* an object. Grasping results in holding the object if the monkey and object are in the same place at the same height.
(a) (4%) Write down the initial state description.
(b) (6%) Write down STRIPS-style definitions of the six actions.

4 (8%) Consider the following problem: A patient arrives at the doctor's office with symptoms that could have been caused either by dehydration or by disease *D* (but not both). There are two possible actions: *Drink*, which unconditionally cures dehydration, and *Medicate*, which cures disease *D*, but has an undesirable side-effect if taken when the patient is dehydrated. Write the problem description in PDDL (Planning Domain Definition Language), and diagram a sensorless plan that solves the problem, enumerating all relevant possible worlds.

5 (6%) Suppose we generate a training set from a decision tree and then apply decision-tree learning to that training set. Is it the case that the learning algorithm will eventually return the correct tree as the training set size goes to infinity? Why or why not?

6 For each of the following determinations, write down the logical representation and explain why the determination is true (if it is):
(a) (3%) For a given program, input determines output.
(b) (3%) Climate, food intake, exercise, and metabolism determine weight gain and loss.

7 (8%) Construct by hand a neural network that computes the XOR function of two inputs. Make sure to specify what sort of units you are using.