

Assignment 4

Total marks: 65.

Out: December 5

Due: December 20

Note: Your report for this assignment should be the result of your own individual work. Take care to avoid plagiarism (“copying”). You may discuss the problems with other students, but do not take written notes during these discussions, and do not share your written solutions.

1. Answer the 4 questions in the Exercises at the end of Chapter 14 (pp. 301-303) for the first “Pots of Water” example on p. 301 in the textbook. [20 points]
2. Answer the 7 questions in the Exercises at the end of Chapter 15 (pp. 323-325) for the first “Pots of Water” example in the textbook. For question 7, note that in the “Pots of Water” puzzle, actions are always executable; find a heuristic that could nonetheless be used to eliminate certain actions from consideration during planning without affecting the ability of the planner to find a plan that achieves the goal when one exists. [30 points]
3. Exercise 4 of Chapter 16 in the textbook. [15 points]

Bonus Problem [10 points]

- a) Write Prolog versions of the precondition axioms, successor state axioms, and initial state axioms for the “Pots of Water” example of Chapter 15.
- b) Write a Golog program that represents a plan to solve the example. Then, use the Golog interpreter on the course web site to run the program and confirm that the goal holds in the situation where the program terminates. That is, show that the query `?- do(your_program, s0, S), contains(pot2, 1, S).` succeeds with an appropriate binding for the situation *S*.
- c) You can define the iterative deepening planning procedure described in Section 15.3.2 of the textbook in Golog as follows:

```

proc(idPlan(N), idPlan(0,N)).

proc(idPlan(M,N), dfPlan(M) #
      ?(M < N) : pi(m1, ?(m1 is M + 1) : idPlan(m1,N))).

proc(dfPlan(N),
      ?(goal) #
      ?(N > 0) : pi(a, ?(acceptable(a)) : a) :
      pi(n1, ?(n1 is N - 1) : dfPlan(n1))).

```

Define the goal fluent `goal(s)` and the forward filtering fluent `acceptable(a,s)` appropriately, and show that this planning procedure can find a solution to the “Pots of Water” example. That is, show that the query `?- do(idPlan(5),s0,S).` succeeds with a binding for the situation `S` that is a solution to the example.