

Programming Assignment 1 – Due October 7

This programming assignment covers Uninformed search, Heuristic search, 2-Person Game Playing and CSP.

[Uninformed Search].

You are given two jugs, a 4-gallon one and a 3-gallon one. Neither has any measuring markers on it. There is a tap that can be used to fill the jugs with water. The goal is to get exactly 2 gallons of water into the 4-gallon jug. Formulate this as a state space search problem and use the following 3 uninformed search strategies – Breadth first, depth first and iterative deepening to find a solution. For each strategy used generate statistics of the time taken to obtain the solution as well as the memory used. Based on your statistics summarize what is the best strategy and why.

[Informed Search].

The sliding-tile puzzle consists of three black tiles, three white tiles, and an empty space. The spaces are arranged in a single row and numbered 1,2,3, . . . ,7. Initially the black pieces occupy spaces 1,2 and 3 while the whites occupy 5,6,7. Space 4 is empty. The puzzle has two legal moves with associated costs: A tile may move into an adjacent empty space. This has a cost of 1. A tile may hop over one or two other tiles into an empty space. This has a cost equal to the number of tiles jumped over. The goal is to have all the white tiles to the left of all the black tiles. The position of the empty space is not important.

Write a A* program to solve this puzzle. Experiment with two different kinds of admissible heuristics. For each heuristic used generate statistics of the time taken to obtain the solution as well as the nodes expanded. Based on your statistics summarize what is the best heuristic and why.

[2-Person Games]

Write a Minimax procedure with alpha-beta pruning to play the tic-tac-toe game. Generate statistics of time used and nodes generated by the Minimax procedure with and without Alpha-Beta pruning. Write a critical analysis.

[CSP]

A Golomb Ruler of order M and length L consists of M marks placed at unit intervals along the ruler such that the differences in spacing between every pair of marks are all distinct. For example the four marks placed at 0, 1, 4 and 6 constitutes a Golomb ruler of order 4 and length 6.

Formulate and implement a CSP solution to verify whether or not a Golomb ruler of a fixed length L for M marks exists. Generate statistics on the number of consistency checks performed for CSP solution with plain Backtracking (BT), BT+MRV, BT+MRV+Forward Checking (FC), BT+MRV+Constraint Propagation. Analyze the impact of these 4 heuristics on scalability of the problem size, i.e., how large a L are you able to handle.

Submission:

On the due date you should email to the TA a zip file containing:

- Source code with good documentation
- A trace of the execution of each program
- Test set
- A report with the required statistics generated for each problem

You have until midnight of the due date to email the zip file.

You should sign up for a demo with a TA. On the demo date you will be given your source files that you should compile and demo to the TA.