Regular Discussion 1 Solutions

Course Overview

Here are some questions for you:

1. What is AI?

2. What can AI do?
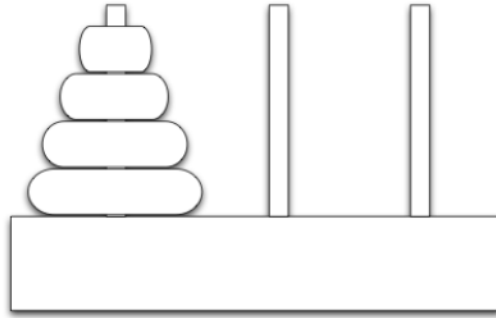
3. What do you want to learn from this course?

There are two types of discussion sections:

1. Regular Discussion

2. Exam Prep

There are 5 graded components:

1. Programming Assignments (25%)

2. Electronic Homework Assignments (10%)

3. Written Homework Assignments (10%)

4. Midterm (20%)

5. Final exam (35%)

1 Towers of Hanoi

The Towers of Hanoi is a famous problem for studying recursion in computer science and recurrence equations in discrete mathematics. We start with N discs of varying sizes on a peg (stacked in order according to size), and two empty pegs. We are allowed to move a disc from one peg to another, but we are never allowed to move a larger disc on top of a smaller disc. The goal is to move all the discs to the rightmost peg (see the figure above).

In this problem, we will formulate the Towers of Hanoi as a search problem.

(a) Propose a state representation for the problem.

One possible state representation would be to store three lists, corresponding to which discs are on which peg. If we assume that the N discs are numbered in order of increasing size $1, ..., n$, then we can represent each peg as an ordered list of integers corresponding to which discs are on that peg.

(b) What is the size of the state space? If there are k pegs and n disks, then the size of the state space is k^n. For this setup, the size is 3^N.

(c) What is the start state? $([1, ..., n], [], [])$

(d) From a given state, what actions are legal?

We can pop the first integer from any list (i.e., peg) and push it onto the front of another list (peg), so long as it is smaller than the integer currently at the front of the list being pushed to (i.e., peg being moved to). Integers can also always be moved to any empty list.

(e) What is the goal test? Is the state the same as $([], [], [1, ..., n])$?

2 Search Algorithms in Action (Uninformed Search Review)
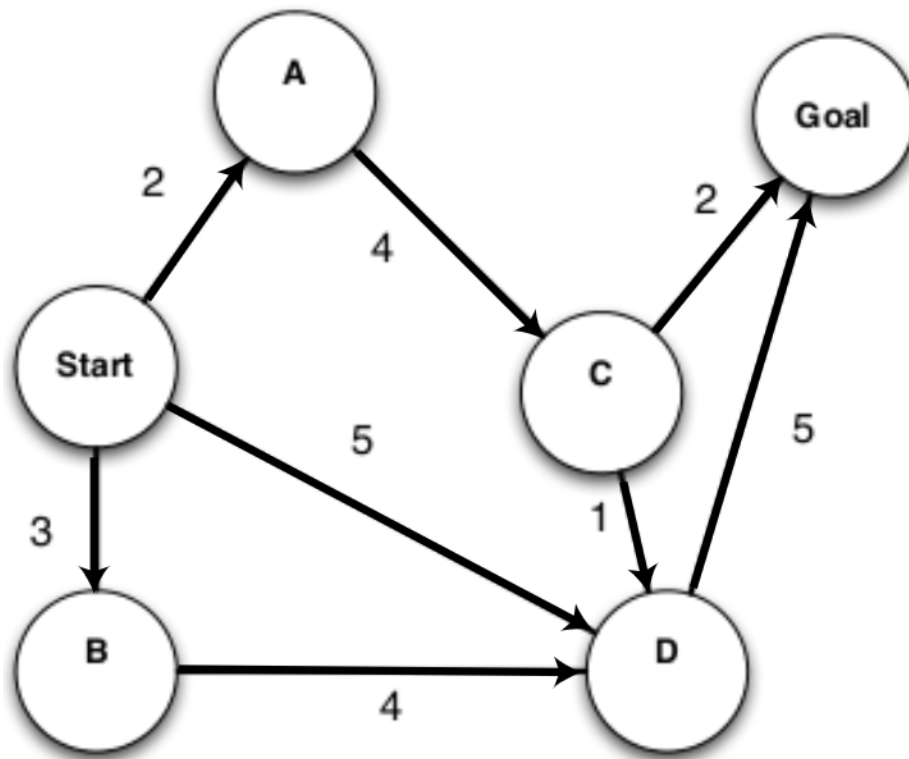
For each of the following graph search strategies, work out the order in which states are expanded, as well as the path returned by graph search. In all cases, assume ties resolve in such a way that states with earlier alphabetical order are expanded first. Remember that in graph search, a state is expanded only once.

a)   Depth-first search.
States Expanded: Start, A, C, D, Goal
Path Returned: Start-A-C-D-Goal

b)   Breadth-first search.
States Expanded: Start, A, B, D, C, Goal
Path Returned: Start-D-Goal

c)   Uniform cost search.
States Expanded: Start, A, B, D, C, Goal
Path Returned: Start-A-C-Goal