CS188: Artificial Intelligence, Spring 2009
Written Assignment 1: Search and CSPs

Due: February 10
You can work on this in groups, but everyone should turn in his/her own work.
Don't forget your name and login.

1 More Pancake Heuristics (6 points)

Here, we consider the pancake problem from class (lecture 3). A server is given a stack of n pancakes. Each pancake is a different size. The server can flip the top k pancakes, reversing their order. **The cost of flipping k pancakes is k.** The server's goal is to order the pancakes from smallest (top) to largest (bottom), with minimal cost. More formally, the search states are all permutations σ of $(1, 2, 3, \ldots, n)$, and the goal is $(1, 2, 3, \ldots, n)$. The successor function gives the outcome of flips, for example:

Successors$((3, 4, 1, 2, 5))$ =

action	cost	successor state
flip 2	2	(4,3,1,2,5)
flip 3	3	(1,4,3,2,5)
flip 4	4	(2,1,4,3,5)
flip 5	5	(5,2,1,4,3)

Here are three heuristics for the pancake problem:

1. H_1, The largest pancake that is out of place: largest i such that $i \neq \sigma_i$
2. H_2, The number of pancakes out of position: count of all i such that $i \neq \sigma_i$
3. H_3, One less than the size of the pancake at the top of the stack: $\sigma_1 - 1$

a) Circle all of the following heuristics that are *admissible*:

i.) H_1 ii.) H_2 iii.) $H_1 + H_2$ iv.) $H_2 + H_3$ v.) $\max(H_1, H_2, H_3)$

b) A heuristic H_A dominates a heuristic H_B if $H_A(n) \geq H_B(n)$ for every state. Circle all of the following statements that are true:

i. H_1 dominates H_2

ii. H_1 dominates H_3

iii. H_2 dominates H_1

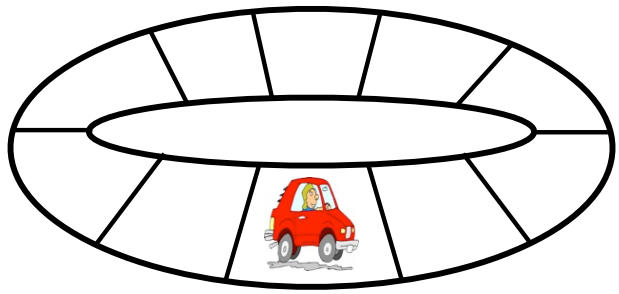
c) Circle all of the following heuristics that are *consistent*:

i.) H_1 ii.) H_2 iii.) H_3 .

2 Driving in Circles (7 points)

A car agent wants to drive k times around a circular track of length L, then park where it started. Its initial speed is 0, but the agent can move at the integer speeds in $[0, V]$. At each time step, the agent can either *coast* (not change speed), *accelerate* (increase speed by 1), or *decelerate* (decrease speed by 1).

Once an action is selected, the agent then moves a number of squares equal to its *NEW* adjusted speed. For example, if the first action is to *accelerate*, the agent will end up one square to the right with a new speed of 1. The agent's goal is to find a plan which parks it (at zero speed) where it began after circling the track k times, using as few actions (time steps) as possible.



a) Describe the state space, start state and goal test for this problem

b) What is the maximum branching factor of this problem? Briefly justify your answer.

c) Is the number of spaces left to go an admissible heuristic? Why or why not?

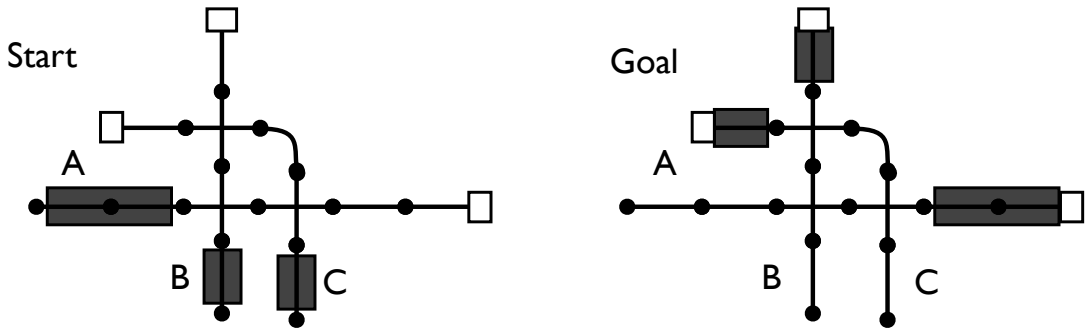
d) State and justify a non-trivial, admissible heuristic which is not the number of spaces left to go.

e) Is breadth-first search guaranteed to be optimal for this problem?

f) Will depth-first search be complete for this problem? Briefly justify your answer.

3 Trains (7 points)

A train scheduler must decide when trains A, B and C should depart. Once a train departs, it moves one space along its track each hour until it arrives at its destination platform. Each train can depart at 1, 2 or 3 pm. The scheduler has two restrictions: All trains must leave at different times, and two trains should not occupy crossing sections of track in the same hour. Note that train A is two spaces long.



a) Describe the constraint satisfaction problem that, when solved, will tell the train scheduler when each train should depart. Let the variables A, B and C represent the departure times of the three trains.

b) Draw the constraint graph for the CSP you defined.

c) After selecting $A = 2$, cross out all values for B and C eliminated by forward checking.

A	B	C
2	1 2 3	1 2 3

d) Cross out all values eliminated by arc consistency before assigning any variables.

A	B	C
1 2 3	1 2 3	1 2 3

e) After selecting $A = 2$, cross out all values for B and C eliminated by arc consistency.

A	B	C
2	1 2 3	1 2 3

f) Describe the execution of backtracking search using forward checking and the minimum remaining values (MRV) and least constraining values (LCV) heuristics. Specifically, in what order are the variables assigned and what values do they take? Start by assigning variable A. You may not need to fill all the lines below:

(1) variable A is assigned value . (2) variable is assigned value .
(3) variable is assigned value . (4) variable is assigned value .
(5) variable is assigned value . (6) variable is assigned value .