# Q1. Search

(a) Consider a graph search problem where for every action, the cost is at least ϵ, with $\epsilon > 0$. Assume the used heuristic is consistent.

(i) [*true* or *false*] Depth-first graph search is guaranteed to return an optimal solution.
False. Depth first search has no guarantees of optimality. Further, it measures paths in length and not cost.

(ii) [*true* or *false*] Breadth-first graph search is guaranteed to return an optimal solution.
False. Breadth first search has no guarantees of optimality unless the actions all have the same cost, which is not the case here.

(iii) [*true* or *false*] Uniform-cost graph search is guaranteed to return an optimal solution.
True. UCS expands paths in order of least total cost so that the optimal solution is found.

(iv) [*true* or *false*] Greedy graph search is guaranteed to return an optimal solution.
False. Greedy search makes no guarantees of optimality. It relies solely on the heuristic and not the true cost.

(v) [*true* or *false*] A* graph search is guaranteed to return an optimal solution.
True, since the heuristic is consistent in this case.

(vi) [*true* or *false*] A* graph search is guaranteed to expand no more nodes than depth-first graph search.

False. Depth-first graph search could, for example, go directly to a sub-optimal solution.

(vii) [*true* or *false*] A* graph search is guaranteed to expand no more nodes than uniform-cost graph search.
True. The heuristic could help to guide the search and reduce the number of nodes expanded. In the extreme case where the heuristic function returns zero for every state, A* and UCS will expand the same number of nodes. In any case, A* with a consistent heuristic will never expand more nodes than UCS.

(b) Let $h_1(s)$ be an admissible A* heuristic. Let $h_2(s) = 2h_1(s)$. Then:

(i) [*true* or *false*] The solution found by A* tree search with h_2 is guaranteed to be an optimal solution.

False. h_2 is not guaranteed to be admissible since only one side of the admissibility inequality is doubled.

(ii) [*true* or *false*] The solution found by A* tree search with h_2 is guaranteed to have a cost at most twice as much as the optimal path.
True. In A* tree search we always have that as long as the optimal path to the goal has not been found, a prefix of this optimal path has to be on the fringe. Hence, if a non-optimal solution is found, then at time of popping the non-optimal path from the fringe, a path that is a prefix of the optimal path to the goal is sitting on the fringe. The cost $\bar{g}$ of a non-optimal solution when popped is its f-cost. The prefix of the optimal path to the goal has an f-cost of $g + h_0 = g + 2h_1 \leq 2(g + h_1) \leq 2C^*$, with C^* the optimal cost to the goal. Hence we have that $\bar{g} \leq 2C^*$ and the found path is at most twice as long as the optimal path.

(iii) [*true* or *false*] The solution found by A* graph search with h_2 is guaranteed to be an optimal solution.

False. h_2 is not guaranteed to be admissible and graph search further requires consistency for optimality.

(c) The heuristic values for the graph below are not correct. For which single state (S, A, B, C, D, or G) could you change the heuristic value to make everything admissible and consistent? What range of values are possible to make this correction?

State: B   Range: [2,3]
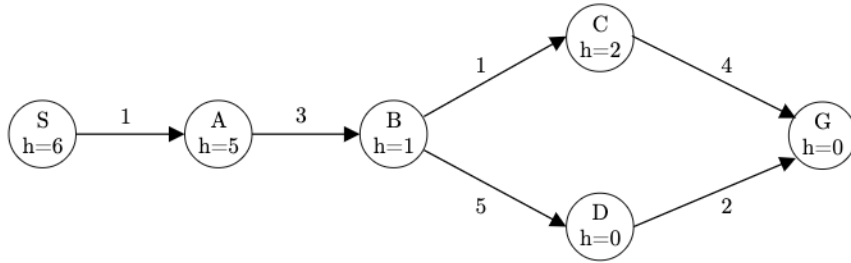
Q2. Power Pellets

Consider a Pacman game where Pacman can eat 3 types of pellets:

- Normal pellets (n-pellets), which are worth one point.

- Decaying pellets (d-pellets), which are worth $max(0, 5 - t)$ points, where t is time.

- Growing pellets (g-pellets), which are worth t points, where t is time.

The pellet's point value stops changing once eaten. For example, if Pacman eats one g-pellet at $t = 1$ and one d-pellet at $t = 2$, Pacman will have won $1 + 3 = 4$ points.

Pacman needs to find a path to win at least 10 points but he wants to minimize distance travelled. The cost between states is equal to distance travelled.

(a) Which of the following must be including for a minimum, sufficient state space?

- ■ Pacman's location
- ☐ Location and type of each pellet
- ☐ How far Pacman has travelled
- ■ Current time
- ☐ How many pellets Pacman has eaten and the point value of each eaten pellet
- ■ Total points Pacman has won
- ■ Which pellets Pacman has eaten

A state space should include which pellets are left on the board, the current value of pellets, Pacman's location, and the total points collected so far. With this in mind:
(1) The starting location and type of each pellet are not included in the state space as this is something that does not change during the search. This is analogous to how the walls of a Pacman board are not included in the state space.
(2) How far Pacman has travelled does not need to be explicitly tracked by the state, since this will be reflected in the cost of a path.
(3) Pacman does need the current time to determine the value of pellets on the board.
(4) The number of pellets Pacman has eaten is extraneous.
(5) Pacman must track the total number of points won for the goal test.
(6) Pacman must know which pellets remain on the board, which is the complement of the pellets he has eaten.

(b) Which of the following are admissible heuristics? Let x be the number of points won so far.

- ■ Distance to closest pellet, except if in the goal state, in which case the heuristic value is 0.
- ☐ Distance needed to win $10 - x$ points, determining the value of all pellets as if they were n-pellets.
- ☐ Distance needed to win $10 - x$ points, determining the value of all pellets as if they were g-pellets (i.e. all pellet values will be t.)
- ☐ Distance needed to win $10 - x$ points, determining the value of all pellets as if they were d-pellets (i.e. all pellet values will be $max(0, 5 - t)$.
- ☐ Distance needed to win $10 - x$ points assuming all pellets maintain current point value (g-pellets stop increasing in value and d-pellets stop decreasing in value)
- ☐ None of the above

(1) Admissible; to get 10 points Pacman will always have to travel at least as far as the distance to the closest pellet, so this will always be an underestimate.
(2) Not admissible; if all the pellets are actually g-pellets, assuming they are n-pellets will lead to Pacman collecting more pellets in more locations, and thus travel further.

(3) Ambiguous; if pellets are n-pellets or d-pellets, Pacman will generally have to go further, except at the beginning of the game when d-pellets are worth more, in which case this heuristic will over-estimate the cost to the goal. However, if Pacman is allowed to stay in place with no cost, then this heuristic is admissable because the heuristic will instead calculate all pellet values as 10. This option was ignored in scoring.
(4) Not admissible; if pellets are n-pellets or g-pellets, Pacman would have an overestimate.
(5) Not admissible; if pellets are g-pellets, then using the current pellet value might lead Pacman to collect more locations, and thus travel further than necesarry.

(c) Instead of finding a path which minimizes distance, Pacman would like to find a path which minimizes the following:

$$C_{new} = a*t + b*d$$

where t is the amount of time elapsed, d is the distance travelled, and a and b are non-negative constants such that $a+b=1$. Pacman knows an admissible heuristic when he is trying to minimize time (i.e. when $a=1, b=0$), h_t, and when he is trying to minimize distance, h_d (i.e. when $a=0, b=1$). Which of the following heuristics is guaranteed to be admissible when minimizing C_{new}?

☐ $mean(h_t, h_d)$ ■ $min(h_t, h_d)$ ☐ $max(h_t, h_d)$ ■ $a*h_t + b*h_d$
☐ None of the above

For this question, think about the inequality $C_{new} = a*t + b*d \geq a*h_t + b*h_d$. We can guarantee a heuristic h_{new} is admissible if $h_{new} \leq a*h_t + b*h_d$
(1) If $a=b$, $0.5*h_t + 0.5*h_d$ is not guaranteed to be less than $a*h_t + b*h_d$, so this will not be admissible.
(2) $min(h_t, h_d) = a*min(h_t, h_d) + b*min(h_t, h_d) \leq a*h_t + b*h_d$
(3) $max(h_t, h_d)$ will be greater than $a*h_t + b*h_d$ unless $h_t = h_d$, wo this will not be admissible.
(4) Admissible.