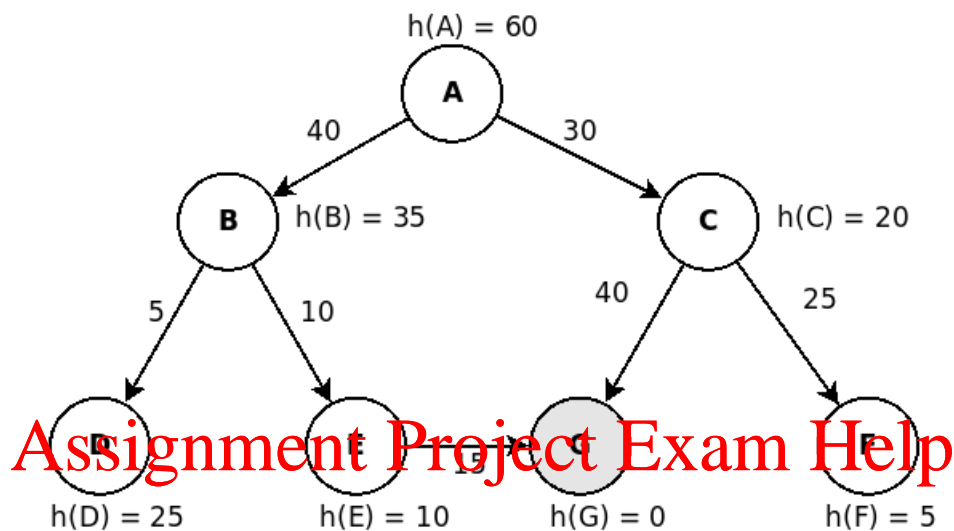**Monash University   Faculty of Information Technology**
FIT3080 Week 4 Lab 3: Informed Search

Exercise 1: *Algorithm A*

Consider the (full) state space below. Indicate (1) the order in which nodes are expanded, and (2) the nodes remaining in memory after finding the goal, for the following search strategies (assume typical left-to-right operator tie breaking):

(a) Depth-First Iterative Deepening

(b) Greedy Best-First Search

Let's consider a new strategy called Algorithm A. This algorithm can be described as a best-first search which uses the following expansion priority for a node n:

$$\mathbf{f(n) = g(n) + h(n)}$$

As usual, $g(n)$ denotes the cost-so-far (i.e., from the start node to node n) and $h(n)$ is an estimate of the cost-to-go (i.e., from node n to the goal node). We can instantiate Algorithm A as using the Graph-search or Tree-search framework, provided the following invariant properties for the $g-$ and $h-$value function are true:

$$g(n) >= g^*(n)$$
$$h(n) >= 0$$

(c) Instantiate Algorithm A (as Graph or Tree- search) to solve the above problem.

(d) Is algorithm A the same algorithm as A*?

(e) How can we modify things (or what can we modify) so that Algorithm A behaves like A*?

(f) After the modification, what nodes are expanded by A*?

Exercise 2: *Algorithm $A^{(*)}$ – Traveling Salesman Problem*

A salesperson must visit each of n cities exactly once. Assume that there is a road between each pair of cities. Starting at city A, find the route of minimal distance that visits each of the cities only once and returns to city A.

(a) Propose a state space and action space for this problem, explaining clearly under what state conditions certain actions are allowed.

(b) Propose two (non-zero) h functions for this problem. Is either of these h functions admissible (a lower bound of h^*)?

(c) Apply algorithm A with one of these h functions to the following 5 city problem:
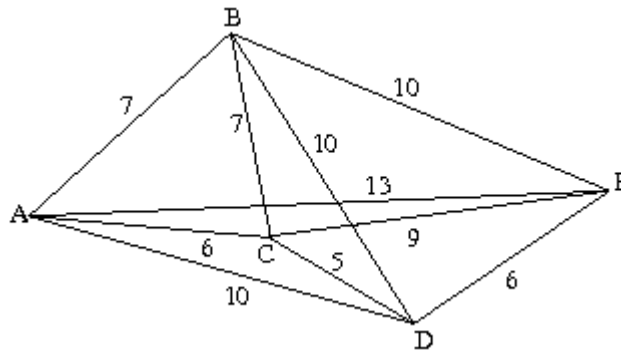
Exercise 3: Algorithm $IDA^{(*)}$ – 5-puzzle problem

Consider a problem called the 8-puzzle. The problem has the start and goal state as follows. Throughout the question, when doing a search, give value of $f(n), g(n)$ and $h(n)$ at each node n. The estimated cost function $g(n)$ is the number of steps from the initial node.

(a) Construct two non-trivial heuristic functions, h_1 and h_2, that you think may help the algorithm to quickly find a solution. Are these functions admissible? Explain why?

(b) Which heuristic function is more efficient? Explain why?

(c) Use h_1 to do an IDA* search to get from the start state to the goal state in the minimum number of steps. Make sure to show all working including value of f, h, g functions in each steps.

(d) (optional/homework) Use h_2 to do an IDA* search to get from the start state to the goal state in the minimum number of steps. Make sure to show all working including value of f, h, g functions in each steps.