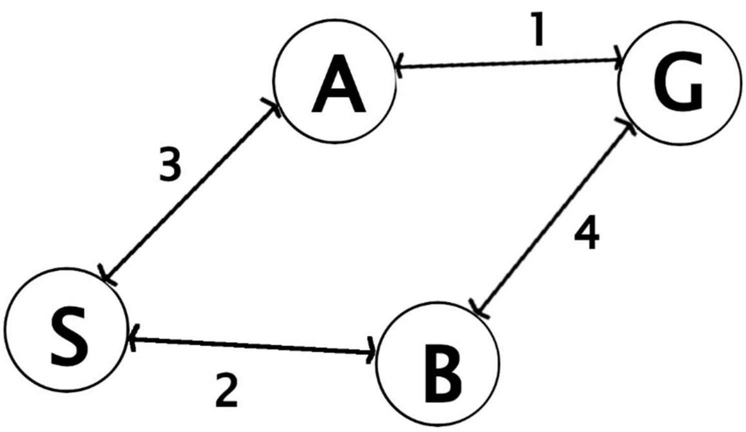
Introduction to Artificial Intelligence
Assignment 01

Task 1

Upon finding a path that leads to a goal node, there still might exist the possibility of finding a path with less costs. So even while another path than the premature solution might not yet lead to the goal node, as long as its costs are less than the costs of the premature solution, it could still be a more cost-efficient solution.

Example



1. (((S).0))
2. (((B S).2) ((A S).3))
3. (((A S).3) ((G B S).6))
4. (((G A S).5) ((G B S).6))

Had UCS, as described in the task, stopped when it first encountered a path terminating at the goal node during the expansion step, it would have terminated upon finding the solution

((G B S).6). So instead of reaching step 4 and finding the more cost-efficient solution ((G A S).5), UCS would have just stopped in step 3 after having found the first solution leading to a goal node, namely ((G B S).6).

Task 2

"Proof" that the heuristic implemented is admissible:

Admissibility is given if a heuristic function $\hat{h}$ never overestimates the distance to the nearest goal state. Minimum number of moves[1] required to solve Towers of Hanoi: 2^{n-1}. Finding an admissible heuristic could look as follows:

> Find out how many discs are not yet in order
>
> Use the equation with which one can determine the minimum number of moves required to solve Towers of Hanoi in order to find the smallest upper bound for the heuristic
>
> Decrease the smallest upper bound, i.e. through decrementing it by 1, in order to obtain a heuristic that always underestimates the distance to the nearest goal

By using a number smaller than the minimum number of moves required to solve the Towers of Hanoi for the heuristic, it's practically impossible to overestimate the distance to the nearest goal state.

---

[1] Hinz et al., 2013, The Tower of Hanoi – Myths and Maths, Birkhäuser, Basel