Assignment-1 :: Artificial Intelligence

IMPORTANT

For each individual question, submissions should include:

- A single .py file (preferred) of the executable code; in case you have multiple files, make sure to add a README file with necessary instructions for execution. The code(s) will be executed during evaluation. If the reported output does not match with the output produced during evaluation, the score will be deducted. **Make sure that your code does not contain unnecessary comments, similar to a code produced by LLM.**
- A report containing the explanations, input used, output, and necessary discussions and screenshots which can support your solution. If your code is large and/or highly modularized, an explanation on the tasks performed by individual functions/classes should be added under a separate section. **This is a coding assignment; thus, adding unnecessary theory and/or texts just to make it look comprehensive will be strictly penalized.**

*** Plagiarism will not be tolerated. If caught, zero will be awarded.**

Q1. Consider a simple N-Queens problem. Take a suitable value of N and write a Python code for simulated annealing to solve the problem. As we know, if we take T=0 in SA, it falls back to a simple hill-climbing search. Modify the SA algorithm in order to make it an HC solution. In the report, write proper theoretical justification in support of your modification (e.g., calculation of the acceptance probability) and the behavioural difference between the two approaches. (Note: No marks will be given for simply implementing the HC solution.)

Q2. Formulate the 8-puzzle as a state-space search problem by specifying the state representation, goal state, actions, and path cost function. Your start state should be a random board initialization (other than the goal state, shown below). 'B' denotes the blank cell.

Goal state:

1 2 3
4 5 6
7 8 B

Implement A* search by taking the heuristic functions as,

$h_1(n)$ = No. of misplaced tiles

$h_2(n)$ = Total Manhattan distance