Name : Alok Bhawarkar
Roll : PA06
Panel : 1

Page No.:

AI

AI ASSIGNMENT - 1

- Title : Implementation of A* algorithm for 8 puzzle problem.

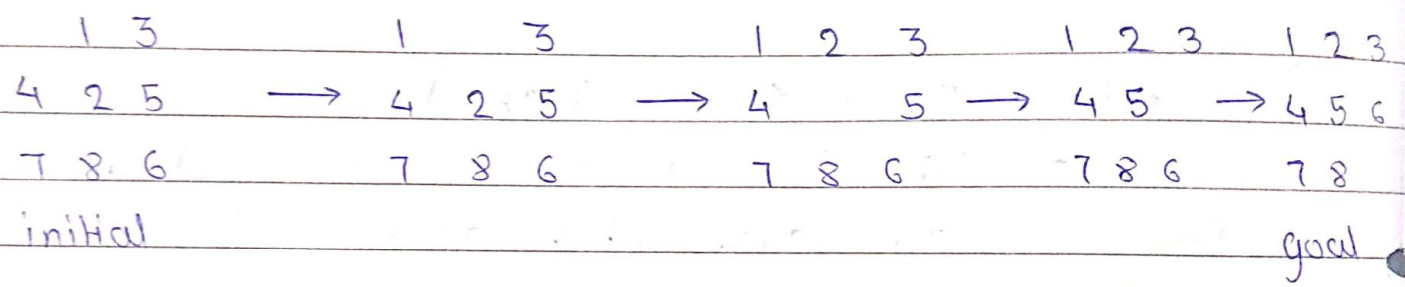
- Aim : Solve 8 puzzle problems using A* algorithm

- Objective : To study and implement A* algorithm for 8 puzzle problem.

- Theory :
  - Best First Search Method
    - The idea is to use an evaluation function to decide which adjacent is most promising and then explore Best First Search falls under the category of Heuristic Search or Informed Search.
    - It is the combination of depth - First search and breadth First search.
    - Worst case time complexity - O(n * log n)
      n → number of nod

  - Or Graph
    - A form of a graph or tree used in problem solving and problem decomposition. The nodes of the graph represent states or goals and their accessories are labelled as either AND or OR branches. OR branches indicates alternative subgoals, any one of which could satisfy the parent goal.

- 8 Puzzle Problem
  - It is played on a 3×3 grid with 8 squares blocks labeled 1 to 8 and a blank square.
  - Your goal is to rearrange the blocks so that they are in order. You are permitted blocks horizontally or vertically into the blank square
  - eg :



  - We used A* algorithm to solve the 8 puzzle problem.

- DATA STRUCTURE & OTHER DETAIL ABOUT A* ALGORITHM
  - A* is best first search algorithm

$$f(n) = g(n) + h'(n)$$

$g(n)$ = Sum of edge costs from start to n
$h'(n)$ = estimate of lower cost from h to goal.
$f(n)$ = actual distance so far + estimate distance
$h(n)$ = is said to be admisible if it underestimates the cost of solution that can be reached from n
$f(n)$ is cost of cheapest solution put n from n to goals and if h' is admissible

$$h'(n) \leq c(n)$$

We can prove that if n'(n) is admissible then search will find an optimal solution.

A* Properties
- It is admissible.
- Solution found by A* is optimal solution.
- No. of nodes A searched still exponential is worst case otherwise heuristic is logarithimically very accurate.

- INPUT - Initial state
- OUTPUT - Solution / goal state with optimal path
- PLATFORM - Linux
- FAQ's

1. What is heuristic function ? Write it's advantage ?

ans:-
- It is a function that reaches alternatives in search algorithms at each branching step based on available information to decide which branch to follow.
- Heuristic are used to identify most promising search.
- denoted by n(n)
- n(n) - estimated cost of cheapest path from node n to goal node.

- Advantages
  - It can provide some quick and recursively in expensive feedback to designers.
  - You can obtain feedback early in design process
  - Assigning correct heuristic can help suggest the best corrective measure.

2. Explain A* Algorithm with example ?

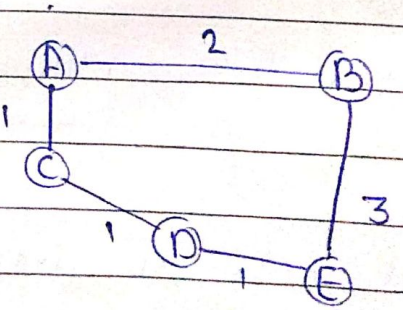
ans: A* algorithm has 3 parameters -

1. g - the cost of moving from initial cell to the current cell. Basically it is sum of all cells visited since leaving first cell.
2. h - (Heuristic value) - Estimated cost of moving from current cell to final cell. It is estimated cost.
3. f - sum of g & h

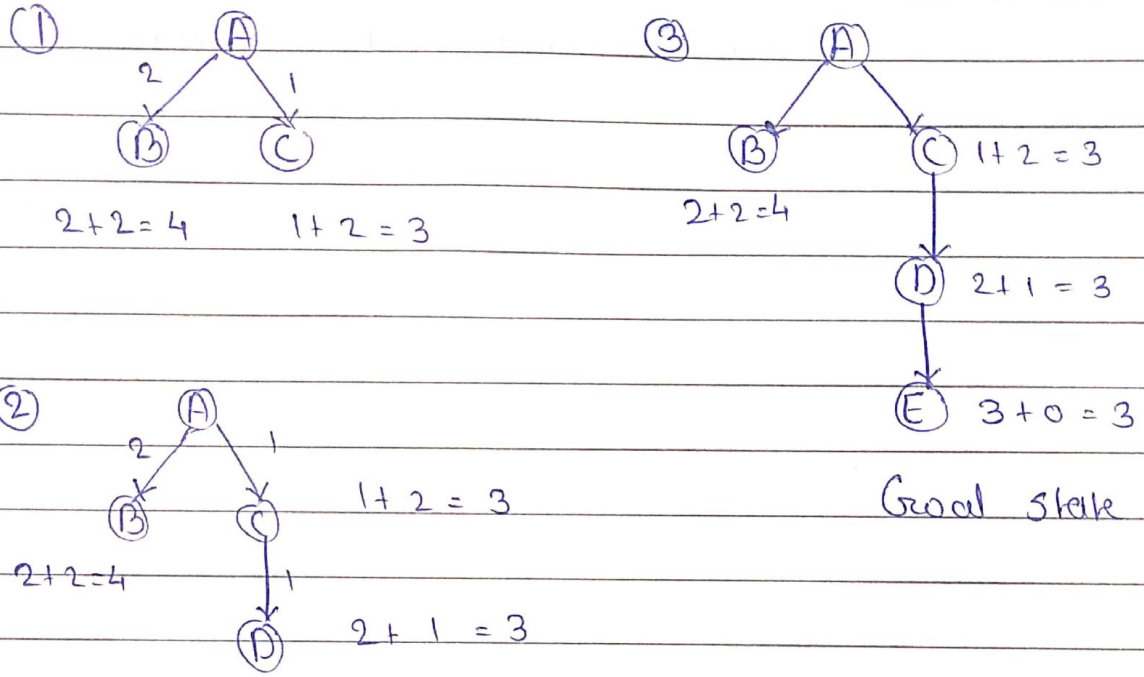
   f = g + h

The way algorithm matches its decision is by taking f-value into account. The algorithm selects the smallest f valued cell and moves to that cell.

example. :



Suppose you have following graph. Initial node is A & goal is e.
At every step f value is being re-calculated by adding together the g & h values. The minimum f values node is selected to reach goal state.



3 Explain different heuristic function that can be used for 8 puzzle problem ?

ans: 1. n - max swap :
Assume you can swap any title with the space use the number of steps it takes to solve problem as heuristic value.

2. x - y :
Decompose problem into two one dimensional where the "space" can swap with any title in adjacent new column.

3. Wilkson's sequence score:

$h(n) = p(n) + 3\,s(n)$

$p(n)$ → Md.hattan distance of each HiHe from its original position.

$s(n)$ → sequence score obtained by checking around non-central square.