**CS 401 ARTIFICIAL INTELLIGENCE SECTION B**

**MIDTERM 1 SOLUTIONS**

**QUESTION 1 (Marks 8+2)**

Apply iterative deepening search (tree search) to the following problem. Initial node is B and final node is E. The cost of each link is 1 as indicated.

a. Show the search trees that develop. When making the branches, order the nodes in alphabetical order

b. Write down very clearly the order in which nodes are expanded



**SOLUTION**

Depth first search trees are generated for depth limit 0,1,2 till goal is found



**ORDER OF EXPANSION**

B BAD BABCDE

**QUESTION 2 (Marks: 2+1+7)**

Here is a 2D simplified version of the rubik’s cube game. The target of the game is to reach the following position:

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

This next figure shows the initial position of the board. The rest of 4 figures show its division into 4 blocks, each 2x2:

Four basic actions are allowed. 4 2x2 blocks can be rotated right. The order of actions allowed is: block a right, block b right, block c right and block d right. So for example rotating block b right will result in:

rotate ‘b’ right

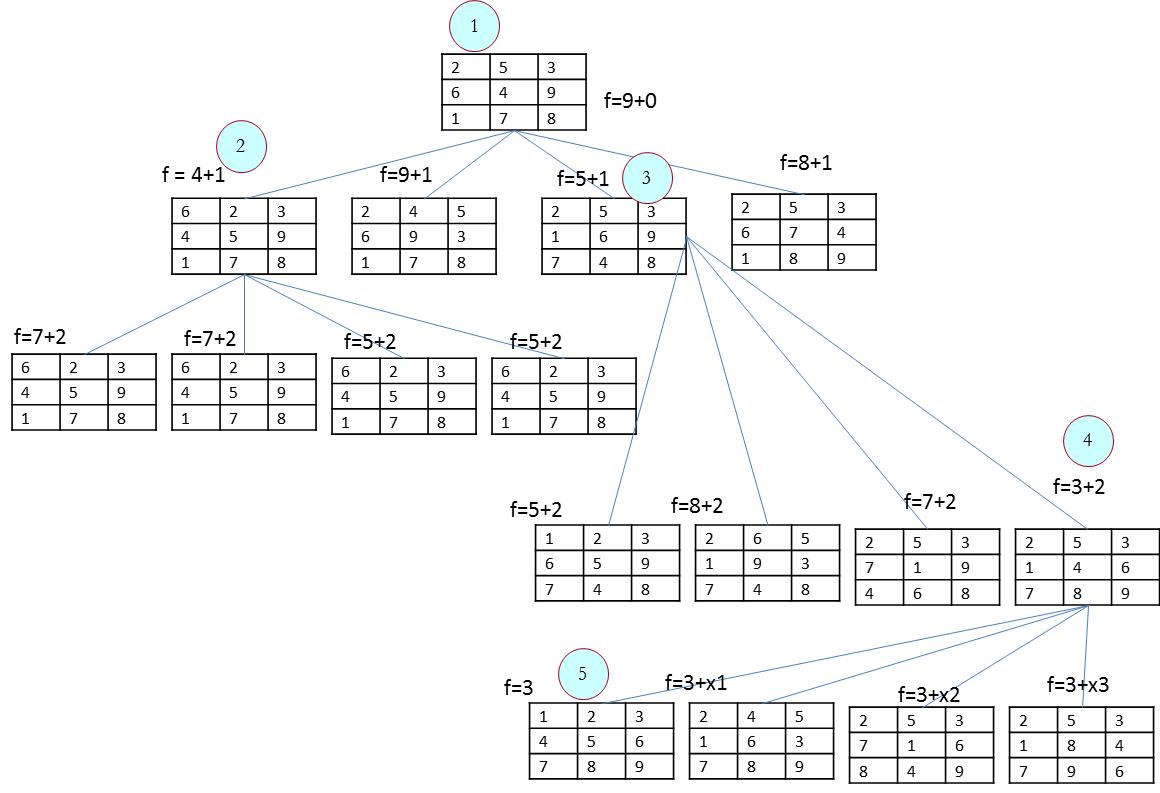
Given an initial configuration of this game:

1. What is the heuristic function for your game?
2. Compute the heuristic function for the initial state.
3. Apply A\* search (graph search) to solve the above. Build the search tree and clearly indicate the order in which each state is expanded.

SOLUTION

There are many possible solutions. One option could be to count the number of wrong cells in a certain block. For example for the initial state: block a has 2 tiles out of place, block b has 3 tiles out of place, block c has 2 and block d has 2 out of place, making it a total of 9

so h the heuristic function for the initial state is 9

****

**Here the order of expansion is shown with the numbers in circles**

**QUESTION 3 (Marks 3+7)**

Here is a game called ‘dots and boxes’. Each player has to make as many boxes as possible. On each turn one player connects two adjacent dots. If the player can complete a box then the same player gets another turn. For example, here is an initial state and corresponding plays by the two players. B is for blue and R is for red

Red turn

Blue turn

You are given the following state: It is B’s turn to play:



a. What is the heuristic for your game? What is the heuristic for the above game position?

b. Make a minmax search tree with a two level lookahead to decide the next turn for B. You can use alpha beta pruning to cut down the unnecessary paths.

**SOLUTION**

There are many possible heuristics which can help blue win. One simple heuristic is to use the following expression:

h(n) = No of squares of blue – no of squares of red

In the figure below h is the heuristic at the leaf node and m is the propagated value of min max score. The x shows where a branch was pruned due to alpha beta pruning. Note: We have ignored all symmetric moves in the figure below. The best option for blue is to follow the first branch.

