**Discrete Mathematics Assignment 2**

* Puzzle Solving Problem, Using AI techniques – Using C++ Language

**INSTRUCTIONS:**

1. This Program needs its user defined datatypes to execute.
2. File named ‘Matrix.h’ and ‘Tree.h’ are its user defined datatypes.
3. File destination of the executable file named ‘Assignment 2 with Matrix and STL.cpp’ and the user defined datatype files: ‘Matrix.h’ and ‘Tree.h’ must be same.
4. The only condition where program will not show the result is, either the given situation is unsolvable, or all the RAM would have been used due to very large amount of data stored in Queue (so as in RAM).

**WORKING OF PROGRAM:**

1. First the program will create the tree; In constructor of the tree it will make a game Board which will contain the final state, used for comparison.
2. Then it will check if the user given board is same as final state; if it is true, then it will simply stop the further program execution as it does not need to be execute.
3. If above is not the case then, it will just add the user given board to the Queue.
4. Now, a board is popped from queue.
5. Create its children.
6. Check each child whether it is same as final or not.
7. If same is found, then it will show all the boards from user given boards to this board.
8. If it is not same, then it will add it to the queue.
9. Step 4 to Step 8 will repeat again and again till final state is obtained.

**PSEUDO CODE:**

Create **tree**;

Take **Head-Node** of **tree** from user;

**if** (**Head-Node** is same as **final state**)

**print** **Head-Node**;

**close** program;

**else**

add **Head-Node** to **Queue**;

**while** (**Node** is not equal to **final state**)

pop **Node** from **Queue**;

create **children** of **Node**;

**if** (any **child** is same as **final state**)

**print** all node form **Head-Node** to **child**;

**else**

add all **children** to **Queue**;