**Lab 5 Solving the 8 puzzle game using search algorithms**

**Experimental purpose:**

1. Review the N puzzle game

2. Master the Breadth First Search, Depth First Search and a Heuristic Based Search (extra)

3. Built your own implementation based on any language you familiar with

**Experimental contents：**

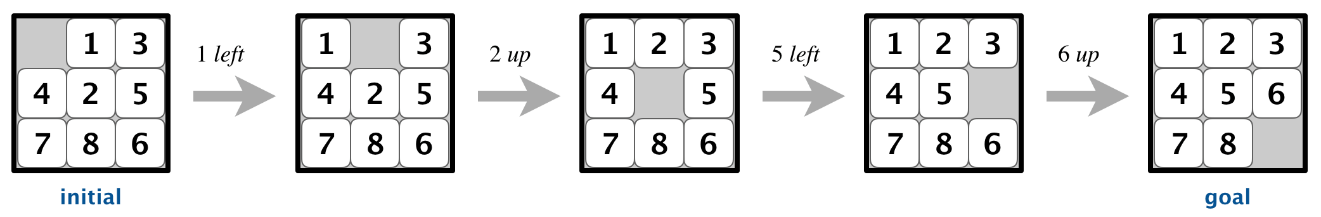
1. Review the 8 puzzle game

2. Study the reference materials

3. Built your own code to solve a random 8 puzzle game

**The 8-puzzle**

The 8-puzzle is a sliding puzzle that is played on a 3-by-3 grid with 8 square tiles labeled 1 through 8, plus a blank square. The goal is to rearrange the tiles so that they are in row-major order, using as few moves as possible. You are permitted to slide tiles either horizontally or vertically into the blank square. The following diagram shows a sequence of moves from an initial board (left) to the goal board (right).

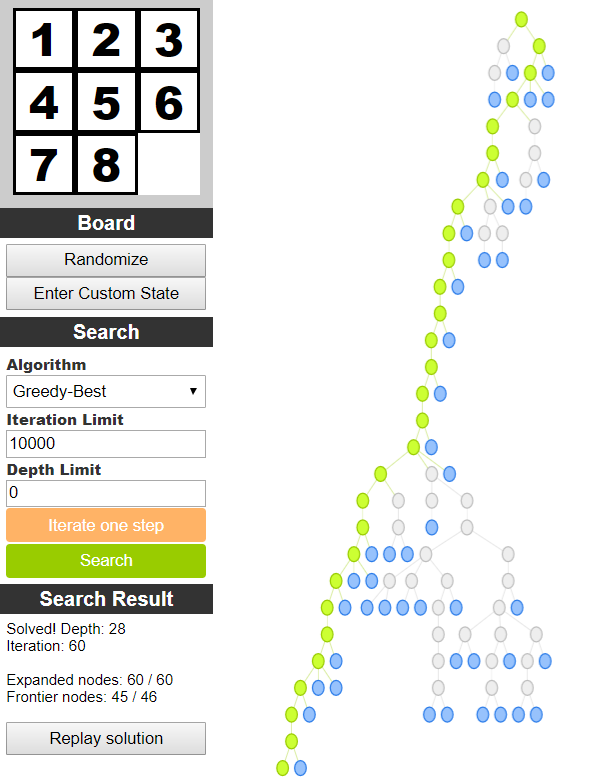


**Reference:**

**1. A web-app that can solve 8-puzzle and visualize the search tree**<https://deniz.co/8-puzzle-solver>  
A web app that solves 8-puzzle game with various tree search techniques. It also visualizes the search tree and supports one-step iterations to explore and comprehend search algorithms.

**Supported search algorithms:**

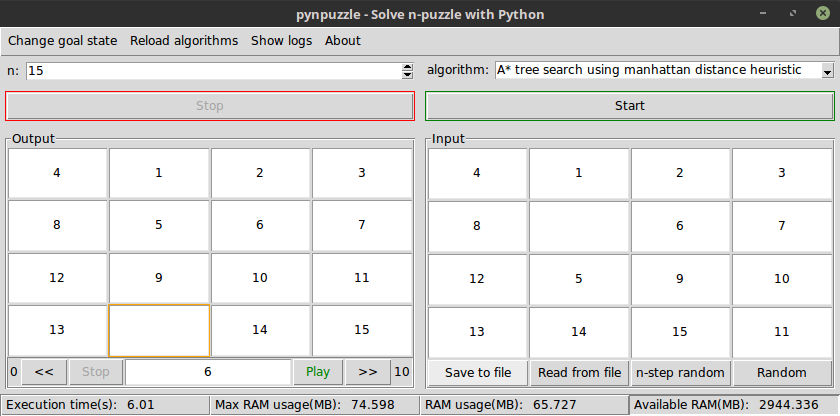
Breadth First  
Uniform Cost  
Depth First  
Iterative Deepening  
Greedy Best  
A\*



**2. “pynpuzzle” (a python implementation)**

https://github.com/mahdavipanah/pynpuzzle

pynpuzzle is an application that helps you to solve n-puzzle problem and also to test your algorithms for n-puzzle problem. It is written in Python and uses tkinter for its graphical interface.



**3. 8puzzle（a python implementation）**

<https://github.com/jmhummel/8puzzle>

Python script for solving the classic "8-puzzle" game

**4. 8 Puzzle Solver Using Search (a JAVA implementation)**

https://github.com/Melvin95/8-Puzzle-Solver-Using-Search?\_pjax=%23js-repo-pjax-container

A program that simulates the solving of the 8 puzzle game using different search algorithms. Breadth First Search, Depth First Search and a Heuristic Based Search(A\* algorithm)

**要求： 3~4人/组，实现8数问题的求解（语言不限）**

**实验报告：英文，手写，报告纸，“目的，内容，算法” ，关键代码或伪码。**