Tutorial 2:
Problem Representation and Uninformed Search

Name	Student no.

For this tutorial, you can discuss the questions in small groups (up to 4 students). Individually submit the answers to each of the 4 Questions.

The idea of the tutorial is not to try and solve the puzzle, but to determine how you will represent this problem so that a computer program can solve the puzzle. You will choose (or be assigned) one of the three problems below.

Question 1

Formally define the problem with the possible states that can occur, the initial state, all of the possible actions available (successor function), goal test, and path cost. Give a brief description for each.

e.g. The 8-puzzle (from p64-65 of Russell and Norvig).

States: *All possible combinations of the location of each of the eight tiles and the blank in one of the nine squares*

Initial state: *Any state can be designated as the initial state*

Possible actions: *Four actions: blank moves Left, Right, Up, or Down*

Goal test: *Checks whether the state matches the goal configuration: tiles are in order 1 to 8 as specified with the blank at the top left corner*

Path cost: *Each step costs 1, so the path cost is the number of steps in the path from the initial state to the goal state*

States:

Initial state:

Successor function:

Goal test:

Path cost:

Question 2

In your group, discuss how to represent this problem in a computer program, including a data structure (e.g. linked list, matrix), a method of representing the possible problem states, and how the transitions between states can be performed. Note that representation refers to the states of the problem, not the entire state space of the problem.

Describe three different ways of representing the problem.

Discuss the advantages and disadvantages of each.

How are the transitions affected by the representation chosen?

What are the constraints on these transitions?

1.

2.

3.

Question 3

Discuss whether the puzzle you have chosen would be better suited to a breadth-first (BFS) or depth-first search (DFS). Consider the completeness, time complexity and space complexity of each method.

Question 4

Write pseudo-code for breadth-first or depth-first search to solve the puzzle. The pseudo-code must be specific to the puzzle, using the appropriate transitions you have previously defined. You should define function names, define input and output parameters for the functions, and write pseudo-code for the functions.

Using,

- *L*, the working list of states to be searched (initially set to the start state)
- *n*, the current state (initially set to the start state)

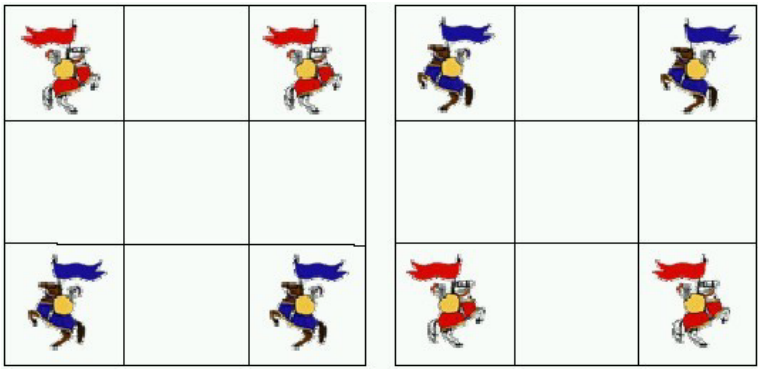
Give

- A goal test (is the current state the goal state?)
- A successor function (what states can you get to from the current state?)
- A process for updating (the transition function i.e. breadth-first or depth-first search – what is the path from the initial state to the goal state?)

The puzzles available are:

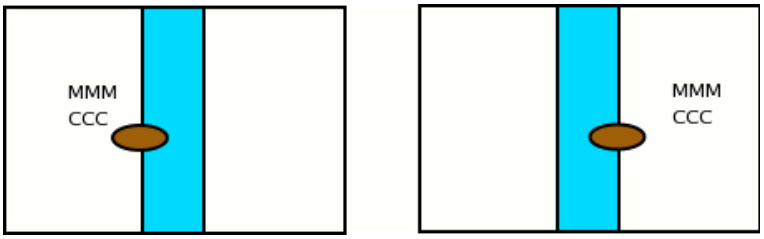
i. *Exchanging the knights:*

Two red knights and two blue knights are placed at the opposite corners of this portion of a chessboard. The blue and red knights need to exchange places (as shown below) by moving according to the rules of chess.



ii. *Missionaries and Cannibals:*

There are three missionaries and three cannibals on one side of a river and they need to get to the other side. There is a two-person boat that can be used to ferry people across. At least one person is needed to row the boat across the river (back and forth) and the cannibals can never outnumber the missionaries on either side of the river.



iii. *Towers of Hanoi:*

There are three pegs and three different sized discs. The discs must move from the first peg to the third peg by moving one disc at a time. A larger disc cannot be placed on top of a smaller disc.