CS3243 Assignment 1

Goh Kheng Xi, Jevan

A0199806L

Q4.

The following assumptions are made when formulating the search problem:

1. The goal state is independent of the pieces’ colours (i.e. colour of the pieces do not matter)
2. Every piece is 4-sided with a maximum of 1 joint per side.
3. There are only two type of joints, external (protruding out) and internal (protruding in); and an external joint will connect with an internal joint and vice versa, but identical joints does not connect (invalid). Flat sides (i.e. sides without joints) do not connect with other joints.
4. A piece is valid only if all four sides have no invalid connections and have a flat side at the boundary of the jigsaw puzzle.
5. Each piece will either have all external or all internal joints but not both (as observed from Figure 4 provided in the problem statement)

We model each jigsaw piece as objects with the following attributes:

1. Type of joints: “external” or “internal”
2. Location of joints: an array consisting of a combination of “up”, “down”, “left” and “right” to represent the locations of the joints for the piece.

We store each piece in a linked list where the index is the unique ID of the piece (0 to n-1), and we enumerate each orientation of the piece (label from 0 to 3)

The state representation of the jigsaw puzzle is a 2d vector (representing the row and column of jigsaw map) of size 2 tuples, where each tuple consists of an integer that corresponds to a unique piece, and another integer that corresponds to the orientation of the piece. The initial state of the jigsaw puzzle is an empty 2d vector and current coordinates of (0, 0) at top left, and the goal state is a m by k 2d vector of size 2 tuples with no repeating piece IDs.

The action is placing an unplaced piece (does not already exist in the current state) in a specific orientation by specifying the piece id and the orientation number and then moving to the next nearest empty spot (i.e. row += 1 and column += 1). For this action, the transition model will push the new piece with a new orientation into the 2d vector and return the new state with the new cost.

If the new piece added is valid, the step cost is 1; if the piece is invalid, the step cost is infinity. The goal test is performed after adding the new piece and updating the total cost of that state. The goal test is checking if current location is at (m-1, n-1) and if the updated cost of the new state is not infinity.

Q5.

S-A-F-D-G