**CECS 545-50 Exercises 6.1, 6.2, 6.3**

**Ryan Schron**

**Exercise 6.1**

To solve the map-coloring problem for Australia, there first are a few problem specific observations to be made. First, SA is the only location central to the other locations, thus limiting possible color assignments. Secondly, T is independent of all locations and can be given any color in any possible solution. With this in mind, it is impossible to solve the problem using two colors, as SA would always be in conflict with some other territory.

For three colors, you can start solving the problem by assigning any of three colors to SA, leaving a choice of 2 colors alternating for the remaining territories (resulting in 2 patterns). This leaves 3\*2\*3 (any color choice is valid for T) = 18 solutions.

For four colors, there exists 96 different solutions. One of four colors will be assigned to SA, then one of the remaining three to WA, leaving 2 colors for NT and so on. When considering that T can be any color, this leaves 4\*3\*2\*4 = 96 solutions.

**Exercise 6.2**

1. When attempting to solve a crossword puzzled using general search, the states should represent the addition of words. Because general search does not constrain the search space, using words would build a search tree of significantly less nodes than using letters, which would exponentially explode the search tree. A depth first search using words would work well for this problem.
2. Using a CSP approach, the variables should be letters. Because words cross and thus must match letters at certain locations, as words are added it limits the search space by imposing constraints thus reducing the time to solve the remainder of the puzzle.

**Exercise 6.3**

1. In solving the rectilinear floor-planning problem, we can first define each rectangle as having four parameters x, y, w, and h where x and y denote the position of a rectangle, w its width, and h the rectangle’s height. Additionally, the larger rectangle that represents the boundaries of the problem can be represented as having width W and height H. In this system, the smaller rectangles would have the constraints:

1. Rect i, x >= 0

2. Rect i, y => 0

3. Rect i, x + Rect i, w <= W

4. Rect i, y + Rect i, h <=H

5. Rect i, x + Rect i, w <= Rect j, w or Rect i, w >= Rect j, w + Rect j, x

6. Rect i, y + Rect i, h <= Rect j, y or Rect i, y >= Rect j, y + Rect j, h

Constraints 1-4 confines the rectangles to fit within the space defined by the larger rectangle, and constraints 5 and 6 ensure that no rectangles overlap one another.

1. The class scheduling problem involves four variables: professors (P), classrooms (R), classes (C), and time slots (T). One formulation can be to rearrange this to have one variable as the class, and the other 3 as domains within that variable, so that they are organized in the form:

C: < P1, P2, …, Pn >, <R1, R2, …, Rm>, < T1, T2, …, Tq >

Additionally, we would have to impose the constraints that a professor cannot be in two classes at the same time, professors are only assigned to appropriate classes, and you cannot use a room for two classes at the same time.

1. A Hamiltonian tour is a cycle in a graph in which each node is visited exactly one time. This can be formulated in the context of a CSP problem by representing each node as a variable, with its connections as domain. Constraints are that a node cannot be visited if it has already been visited.