Sudoku Reading Notes:

Breadth First Search (BFS):

* Broadens the solution space systematically and explores all possibilities that exist at each level.
* Adv- No deadlock
* Dis – lots of memory
* Using BFS to solve Sudoku:
  + Start at empty root node
  + All possible numbers that can be entered in box-0 developed without violating rules; each of these possible numbers entered into single node at level 0.
  + If the next box is a “given”(question) box then the parent node does not need to develop child nodes and all parent nodes have the same child node (the node that contains the numbers in the question box)
  + If the next box is an “input” (answer) box, then develop all possible numbers that can be entered without violating rules.
  + Continue developing all levels until level 80 is reached – answer if series of steps from level 0 to level 80
  + If none of the nodes can reach the 80th level, there is no solution.

Depth Limited Search (DLS):

* Carries out form initial node in depth until most recent is found.
* Adv – quick to reach depth; small memory (only nodes in active path are stored in tree)
* Dis – max level of solution must be known; if depth limit is smaller than solution depth, can’t find solution.
* Using DLS to solve Sudoku:
  + Start empty root node
  + From there, 1 possible number will be developed that can be entered in box 0 (w/o violating rules); this is level-1 child node.
  + If no child nodes can be expanded – all numbers violate rules, then backtrack to parent and develop another child. (basically one child at a time – Depth bitch)
  + If next box is a given box, then the parent node does not need to develop child node and goes directly to the child node containing the Sudoku question and develops other child nodes.
  + Continue until level 80 – answer is steps from level-1 to level-80
  + If there is a backtrack on the root node – no solution

Overall Conclusion:

* DFS is more efficient and faster
* If sudoku has more than one answer BFS will find it
* BFS is more structured and has a systematic search system.
* Looks like we are using just one of the two, whichever we prefer to use