CPSC3750 Assignment 2 written portion
Joshua Enns
SID 001174789

2)
BFS:
Number of nodes explored: 3,696,597
Total time taken: 23.7s
Maximum nodes in frontier: 16,777,216
Largest value of N in one hour: 8
- When N > 8, it fills up my server's RAM, which overflows and fills up the swap, which overflows and then causes linux to automatically kill the process to prevent the server from crashing.

3)
Iterative Deepening
Number of nodes explored: 4,224,684
Total time taken: 20.5s
Maximum nodes in frontier: 4,206,960
Largest value of N in one hour: 9

4)
6 Queens solutions according to iterative deepening algorithm:

| | | | |
|---|---|---|---|
| row0 : 000010<br>row1 : 001000<br>row2 : 100000<br>row3 : 000001<br>row4 : 000100<br>row5 : 010000 | row0 : 000100<br>row1 : 100000<br>row2 : 000010<br>row3 : 010000<br>row4 : 000001<br>row5 : 001000 | row0 : 001000<br>row1 : 000001<br>row2 : 010000<br>row3 : 000010<br>row4 : 100000<br>row5 : 000100 | row0 : 010000<br>row1 : 000100<br>row2 : 000001<br>row3 : 100000<br>row4 : 001000<br>row5 : 000010 |

The algorithm produced four solutions.

5) The main downfall for breadth first search is that it must keep track of all fringe nodes. This leads to an incredible amount of memory usage. Technically, iterative deepening looks at more nodes than breadth first search but it achieves better performance because it clears its memory before analyzing each maximum depth level.
In the 8 queens program tests, the maximum size of the frontier in iterative deepening is a quarter the size of the one in breadth first search (4,206,960 < 16,777,216). Iterative deepening was able to complete the search in 20.5 seconds, whereas breadth first took 23.7 seconds.
Iterative deepening is definitely the faster and more efficient algorithm for finding a solution to the nqueens problem.