Documentation for main.cpp

These are basic uninformed search algorithms used to search through the state space to solve the puzzle. One major issue was that as the state space got bigger, the execution time exponentially increased. Level 3 of the brick game did not finish execution at all. Levels 0, 1,2 finished for all except breadth first search algorithm. The results are piped out and stored in its respective text files with initial state, solved state and the path used to reach the solution state. The algorithms are also timed, which is noted at the end of the text file. For the given script, the main function calls for level1 of the brick game for testing.

```
// Checks for the existence of the Object c in the vector of puzzles and returns true if found
bool check_exists(vector<Puzzle> p, Puzzle c)
```

```
// breadth first search algorithm with path reconstruction
void bfs_path(Puzzle p)
```

```
// depth first search algorithm with path reconstruction
void dfs_path(Puzzle src)
```

```
// facilitating method for iterative deepening dfs
void dls(Puzzle src, int depth, vector<Puzzle> &visited_list)
```

```
// Iterative deepening dfs search algorithm.
void IDDFS(Puzzle p)
```