**Randyll Bearer (Solo), 3-20-2015**

**Project 2 Proposal: Chess, Tues/Wed 1:00pm**

**Introduction:**

For my second project I plan to create a working, two-player Chess game. I’ve grown up playing Chess, and it seems like a good place to start making games (as I plan on doing in the future) so I thought “Why Not?”. Also, Chess and Computer Science have a pretty long history with each other, so I thought I’d keep them together just a little longer. I intend to model the movements of each piece, have them played on a standard Chess board, and to have my program check that each proposed move is legal.

**How I’ll Do It:**

Naturally Computer AI is pretty difficult to create, which is why I decided to make my game two-player. Perhaps for extra credit I will try to implement AI which can pose a challenge, but that will come later. I intend to use arrays to form a matrix to contain my objects (pieces) as well as squares (spaces), and to run methods to check to see if a move is legal. I do not have any plans to implement a User Interface (as we won’t cover them for much later), but I do have a system to display pieces.

Array 1 = r1 k1 b1 qn kn b2 k2 r2

Array 2 = p1 p2 p3 p4 p5 p6 p7 p8

Array 3 = x x x x x [] x x

Of course this interface isn’t ideal, but it should work. The two “[]” below “p6” show the possible moves for that pawn.

Array 4 = x x x x x [] x x

Array 5 = x x x x x x x x

Array 6 = x x x x x x x x

Array 7 = P1 P2 P3 P4 P5 P6 P7 P8

Array 8 = R1 K1 B1 KN QN B2 K2 R2

**Extra Credit:**

1. **Implement some sort of AI.** I would need to research this further as it seems pretty daunting. It wouldn’t be too hard to simply restrict AI moves to spaces where a piece cannot be lost, but it would lead to… whacky behaviors.
2. **A user interface.** It’s not that it is hard to find some public domain Chess sprites (One Google search returned quite a few), it is just that I do not currently know how to implement any form of interface.
3. **A Turn Timer.** This one shouldn’t be too difficult.
4. **Record Keeping.** Store the amounts of moves made, pieces captured, time taken… etc.

**Topics Covered:**

1. **Files:**If the **Record Keeping** extra credit is chosen then I will have to store the data in a file. The game itself should not require the use of files.
2. **Exception Handling:**  I do not want my game to crash, so I will have to handle exceptions at some point in time.
3. **Arrays:** This is how I plan to store my matrix as mentioned above.
4. **Graphical Interface:** If Extra Credit **User Interface** is chosen.