**Chess Engine**

**in JavaScript**

****

**­­­­**

* Ashish Papanai

**Day-1**

**Understanding the basics of chess.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A8 | B8 | C8 | D8 | E8 | F8 | G8 | H8 |
| A7 | B7 | C7 | D7 | E7 | F7 | G7 | H7 |
| A6 | B6 | C6 | D6 | E6 | F6 | G6 | H6 |
| A5 | B5 | C5 | D5 | E5 | F5 | G5 | H5 |
| A4 | B4 | C4 | D4 | E4 | F4 | G4 | H4 |
| A3 | B3 | C3 | D3 | E3 | F3 | G3 | H3 |
| A2 | B2 | C2 | D2 | E2 | F2 | G2 | H2 |
| A1 | B1 | C1 | D1 | E1 | F1 | G1 | H1 |

**Rank**

**Files**

This is a representation of an actual chess board with each Horizontal Row represents a rank.

Representation followed in this engine.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | B1 | C1 | D1 | E1 | F1 | G1 | H1 |
| A2 | B2 | C2 | D2 | E2 | F2 | G2 | H2 |
| A3 | B3 | C3 | D3 | E3 | F3 | G3 | H3 |
| A4 | B4 | C4 | D4 | E4 | F4 | G4 | H4 |
| A5 | B5 | C5 | D5 | E5 | F5 | G5 | H5 |
| A6 | B6 | C6 | D6 | E6 | F6 | G6 | H6 |
| A7 | B7 | C7 | D7 | E7 | F7 | G7 | H7 |
| A8 | B8 | C8 | D8 | E8 | F8 | G8 | H8 |

**Rank**

**Files**

This is a representation of a chess board according to the planned code.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
| 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 |

**Rank**

**Files**

Indexed representation of a chess board

Modified Board representation to prevent the piece from escaping the board

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 |
| 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 |
| 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 |
| 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 |
| 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 |
| 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 |
| 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 |

If any piece reaches the yellow square, We will not allow and register that move.

Reason for adding layer of two squares on top and bottom:

Rook moves in a L shape in horizontal and vertical direction to forbade the horse from moving out of board in vertical direction we add two yellow boxes, this condition is not possible for horizontal motion of Knight, thus we have only one empty row in the left and right side to prevent.

Similarly, to prevent Rook, Bishop and Queen, we have added One layer at vertical and horizontal positions