

Lab -10

Alpha beta for 8 queens

Algorithm:

```
def is_valid (board, row, col):
    for i in range (row):
        if board[i] == col or (board[i] - col) ==
            also (i - row):
            return false
    return true
```

```
def alpha_beta (board, row, alpha, beta, is_max):
    if row == len (board):
        return 1
    if is_max:
        max_score = 0
        for col in range (len board):
            if is_valid (board, row, col):
                board[row] = col
                max_score = alpha_beta (board, row+1,
                    alpha, beta, false)
                board[row] = -1
            alpha = max (alpha, max_score)
            if beta <= alpha:
                break
        return max_score
    else:
        min_score = float (inf)
        for col in range (len board):
            if is_valid (board, row, col):
                board[row] = col
```


$$\text{min_score} = \min(\text{min_score}, \text{alpha_beta}(\text{Grid}, \text{row} + 1, \text{alpha}, \text{beta}, \text{True}))$$
$$\text{bocod}[\text{row}] = -1$$
$$beta = \min(beta, \text{Milestone})$$

1/2 beta \angle = alpha

break

gehen mit glatte

```
def solve8queens():
```

$$b_{\text{pod}} = [-1] \times 8$$
$$\alpha_{\text{ph}} = -\frac{f_{\text{out}}}{f_{\text{in}}} \left(\frac{1}{1 + \frac{f_{\text{out}}}{f_{\text{in}}}} \right)$$
$$\beta = \text{float}(\text{int})$$

beta = floor ()
 return alpha beta (brood, o, alpha, bek, Tre)

Output: Total soln: 6

A piece of lined paper with a red diagonal line running from the bottom left to the top right. There are several small, dark, irregular marks scattered across the page, some of which appear to be the letter 'O' or '0'. There are also some faint, larger, handwritten marks that look like 'O' or '0' in the upper right quadrant. The paper is otherwise blank, with no text or other markings.

8/28/24