Assignment 5 –

package aman;

import java.util.\*;

public class Nqueens {

static int iter = 0;

public static void main(String args[]) {

int n = 5;

char board[][] = new char[n][n];

for(int i = 0; i < n; i++) {

for(int j = 0; j < n;j++) {

board[i][j] = '.';

}

}

long s = System.nanoTime();

Nqueens(board,0);

System.out.println("\nTotal no of solutions: " + iter);

}

static void Nqueens(char[][] board, int row) {

if(row == board.length) {

iter++;

printBoard(board); //Base condition : print board

}

for(int i = 0; i < board.length;i++) {

if(isSafe(board,row,i)) {

board[row][i] = 'Q';

Nqueens(board,row+1);

board[row][i] = '.'; //Backtracking

}

}

}

static void printBoard(char[][]board) {

System.out.println("\n----------Board----------------");

for(int i =0; i < board.length;i++) {

for(int j = 0; j < board.length;j++) {

System.out.print(board[i][j] +" ");

}

System.out.println("");

}

}

static boolean isSafe(char[][]board, int row, int col) {

int i , j;

for(i=row-1; i>=0; i--) {

if(board[i][col] == 'Q') {

return false;

}

}

for(i = row-1, j = col-1; i>=0 && j>=0 ; j--,i--) {

if(board[i][j] =='Q') {

return false;

}

}

for( i = row-1, j = col+1; i>=0 && j < board.length; i--,j++) {

if(board[i][j] == 'Q') {

return false;

}

}

return true;

}

}