#include<bits/stdc++.h>

using namespace std;

vector<int> column,ld,rd; // for bounding col, left diagonal and right diagonal

void nqueen(int row, string asf, int n, vector<vector<int> > &queen){

if(row==n){

cout<<asf<<"."<<endl; // base case

}

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

if(column[col]==0 && rd[row+col]==0 && ld[(row-col)+n-1]==0){

queen[row][col] = 1; // if queen is safe then place queen in that cell

column[col] = 1;

rd[row+col] = 1;

ld[(row-col)+ n-1] = 1;

nqueen(row+1,asf+to\_string(row)+"-"+to\_string(col)+", ", n,queen);

queen[row][col] = 0;

column[col] = 0;

rd[row+col] = 0;

ld[(row-col)+ n-1] = 0;

}

// 0-1, 1-3, 2-0, 3-2, .

}

}

int main(){

int n;

cin>>n;

vector<vector<int> > queen(n, vector<int>(n));

column.resize(n);

ld.resize((2\*n)-1);

rd.resize((2\*n)-1);

nqueen(0,"",n,queen);

}