## HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

**KHOA CÔNG NGHỆ THÔNG TIN 1**

----\*\*\*----



|  |  |
| --- | --- |
| **Lớp (nhóm) :** | E22CQCN04-B (06) |
| **Họ tên:** | 1. Nguyễn Hữu Phúc 2. Nguyễn Văn Tuấn 3. Phí Quốc Tư Sơn |
| **Bài tập số:** | 3 |
| **Môn:** | Toán rời rạc 1 |
| **Giảng viên:** | Nguyễn Thị Mai Trang |
| **Ngày:** | 16/10/2023 |
| **Điểm:** | ……. |

Hà Nội, 2023

## Đề bài: Bài toán n quân hậu

## Trên bàn cờ kích cỡ 𝑛x𝑛, hãy đặt 𝑛 quân hậu mỗi quân trên 1 hàng sao cho tất cả các quân hậu đều không ăn được lẫn nhau.

## Phân tích

## Input: n

## Ouput: liệt kê các trường hợp để xếp được n quân hậu và vị trí của các quân hậu tương ứng với mỗi trường hợp nxn.

## Ý tưởng:

## Mảng flag1, flag2, và flag3 với giá trị 1 đánh dấu việc quân hậu có thể đặt ở các cột, đường chéo xuôi, và đường chéo ngược.

## Hàm Try được gọi đệ quy để đặt từng quân hậu lên bàn cờ. Trong mỗi bước đệ quy, vòng lặp for duyệt qua các cột từ 1 đến n và kiểm tra xem có thể đặt quân hậu tại vị trí (i, j) không.

## Điều này được kiểm tra bằng cách xem flag1[j], flag2[i - j + n], và flag3[i + j - 1] có bằng 1 không. Nếu có thể đặt, quân hậu sẽ được đặt tại vị trí đó và các mảng đánh dấu sẽ được cập nhật.

## Sơ đồ khối

## 

## 

## Cài đặt thuật toán

#include <bits/stdc++.h>

using namespace std;

int n, cnt = 1, flag1[100], flag2[100], flag3[100], res[100];

// Biến n là số quân hậu, cnt là đếm số trường hợp(dùng để in kết quả), 3 mảng flag đánh dấu chưa xuất hiện lần lượt ở cột, đường chéo xuôi, đường chéo ngược và mảng res để lưu kết quả.

void setup()

{

for (int i = 1; i <= 2\*n + 1; i++)

{

flag1[i] = 1;

flag2[i] = 1;

flag3[i] = 1;

}

}

// Hàm setup đánh dấu tất cả giá trị bằng 1, tức là chưa xuất hiện ở cột, đường chéo xuôi, đường chéo ngược.

void Result()

{

cout << "TH " << cnt++ << ": [ ";

for (int i = 1; i < n; i++) cout << res[i] << ", ";

cout<<res[n]<< " ]"<<endl;

for (int i = 1; i <= n; i++)

{

for (int j = 1; j <= n; j++)

if (j == res[i]) cout << "1 ";

else cout << "0 ";

cout << endl;

}

}

// Hàm Result để in kết quả cho mỗi trường hợp.

void Try(int i)

{

for (int j = 1; j <= n; j++)

{

if (flag1[j] && flag2[i - j + n] && flag3[i + j - 1])

{

res[i] = j;

flag1[j] = 0;

flag2[i - j + n] = 0;

flag3[i + j - 1] = 0;

if (i == n)

Result();

else

Try(i+1);

flag1[j] = 1;

flag2[i - j + n] = 1;

flag3[i + j - 1] = 1;

}

}

}

// Quay lui có điều kiện.

int main()

{

cout << "Nhap so quan hau: ";

cin >> n;

setup();

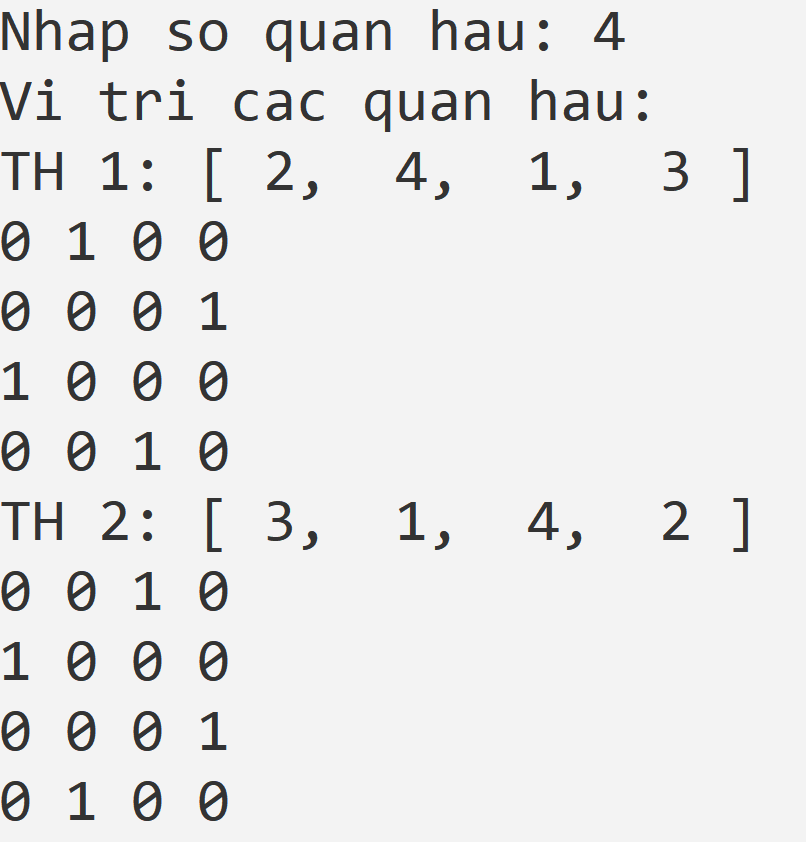
cout << "Vi tri cac quan hau:" << endl;

Try(1);

return 0;

}

## Bộ test

****

