**TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

Logo

Description automatically generated

**TIỂU LUẬN CUỐI KỲ**

**Môn học: Trí Tuệ Nhân Tạo**

**Tên tiểu luận:**

**GIẢI QUYẾT BÀI TOÁN N\_QUEENS BẰNG PHƯƠNG PHÁP HILL CLIMPING**

Giảng viên: PGS.TS. Hoàng Văn Dũng

**Danh sách sinh viên thực hiện**

|  |  |  |
| --- | --- | --- |
| **Mã số SV** | **Họ và tên** | **Mức độ**  **đóng góp (%)** |
| 20110590 | Trần Nguyễn Huy Trường | 100% |
| 20110576 | Nguyễn Nhật Tính | 100% |
| 20110447 | Nguyễn Phước Công | 100% |

*TP. Hồ Chí Minh, tháng .... năm .......*

Catalog

**Mục lục**

[**Phần 1. Mở đầu (mô tả về project) 3**](#_Toc15482)

[**Phần 2. Cơ sở lý thuyết dùng để thực hiện project 4**](#_Toc3304)

[2.1.Các thư viện hỗ trợ lập trình: 4](#_Toc22322)

[2.2.Các phương pháp kỹ thuật được sử dụng: 4](#_Toc6212)

[2.2.1.Đệ quy 4](#_Toc15659)

[2.2.2.Thuật toán Hill Climping 5](#_Toc32338)

[**Mô tả thuật toán và mã giả 5**](#_Toc12987)

[**Phần 3. Phân tích, thiết kế giải pháp 6**](#_Toc22420)

[3.1.Lưu đồ khối của thuật toán 6](#_Toc26046)

[3.1.1 Lưu đồ khối của thuật toán Hill ClimBing 6](#_Toc15443)

[3.1.2 Lưu đồ khối của thuật toán BackTracking 7](#_Toc4039)

[**3.2.Các thuật toán chính: 8**](#_Toc16510)

[**Phần 4. Thực nghiệm, đánh giá, phân tích kết quả 12**](#_Toc23312)

[4.1.Chạy thuật toán Hill Climbing 12](#_Toc15203)

[4.1.1Chạy thuật toán với Random 13](#_Toc10176)

[4.1.2Chạy thuật toán với tùy chọn vị trí 14](#_Toc8792)

[**4.2 Chạy với thuật toán backtracking 15**](#_Toc31865)

[**Phần 5. Phần kết luận 18**](#_Toc13710)

[5.1.Kết quả đạt được 18](#_Toc1107)

[5.2.Hạn chế 18](#_Toc30888)

[5.3.Hướng phát triển 18](#_Toc19884)

[**Tài liệu tham khảo 18**](#_Toc31943)

**Phần 1. Mở đầu (mô tả về project)**

Bài toán xếp quân hậu là bài toán đặt N quân hậu trên bàn cờ vua kích thước N×N sao cho không có quân hậu nào có thể "ăn" được quân hậu khác, hay nói khác đi không quân hậu nào có để di chuyển theo quy tắc cờ vua. Màu của các quân hậu không có ý nghĩa trong bài toán này.

Như vậy, lời giải của bài toán là một cách xếp N quân hậu trên bàn cờ sao cho không có hai quân nào đứng trên cùng hàng, hoặc cùng cột hoặc cùng đường chéo.

# 

# Phần 2. Cơ sở lý thuyết dùng để thực hiện project

-Công cụ lập trình của bài toán được sữ dụng trên Visual studio 2022 trên môi trường .NET và ngôn ngữ lập trình chủ yếu dùng c# winform.

## **2.1.Các thư viện hỗ trợ lập trình:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Diagnostics;

## **2.2.Các phương pháp kỹ thuật được sử dụng:**

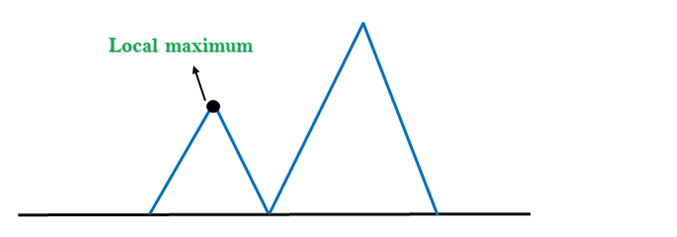
### **2.2.1.Đệ quy**

Đệ quy là phương pháp dùng trong các chương trình máy tính, trong đó có hàm tự gọi chính nó.

Trong toán học và khoa học máy tính, các tính chất (hoặc cấu trúc) được gọi là đệ quy nếu trong đó có một lớp các đối tượng hoặc phương pháp được xác định bằng việc xác định một số rất ít các trường hợp hoặc phương pháp đơn giản (thông thường chỉ một) và sau đó xác định quy tắc đưa ra các trường hợp phức tạp về các trường hợp đơn giản.

Một hàm được gọi là đệ quy nếu bên trong thân hàm có lệnh gọi đến chính nó.

### 2.2.2.Thuật toán Hill Climping



**Mô tả thuật toán và mã giả**

B1: Xét trạng thái đầu: Nếu là đích => dừng

Ngược lại, thiết lập trạng thái bắt đầu = trạng thái hiện tại.

B2: Lựa một luật để áp dụng vào trạng thái hiện tại để sinh ra một trạng thái mới.

B3: Xem xét trạng thái mới này:

Nếu là đích => dừng.

Nếu không phải là đích nhưng tốt hơn trạng thái hiện tại thì thiết lập trạng thái hiệu t là trạng thái mới.

Nếu không tốt hơn thì đến trạng thái mới tiếp theo

Lặp đến khi: gặp đích hoặt không còn luật nào nữa chưa được áp dụng vào trạng thái hiện tại.

### **2.2.3.Thuật toán Backtracking**

Xét tất cả các giá trị X[1] có thể nhận, thử X[1] nhận các giá trị đó. Với mỗi giá trị của X[1] ta sẽ:

Xét tất cả giá trị X[2] có thể nhận, lại thử X[2] cho các giá trị đó. Với mỗi giá trị X[2] lại xét khả năng giá trị của X[3]...tiếp tục như vậy cho tới bước:

...

....

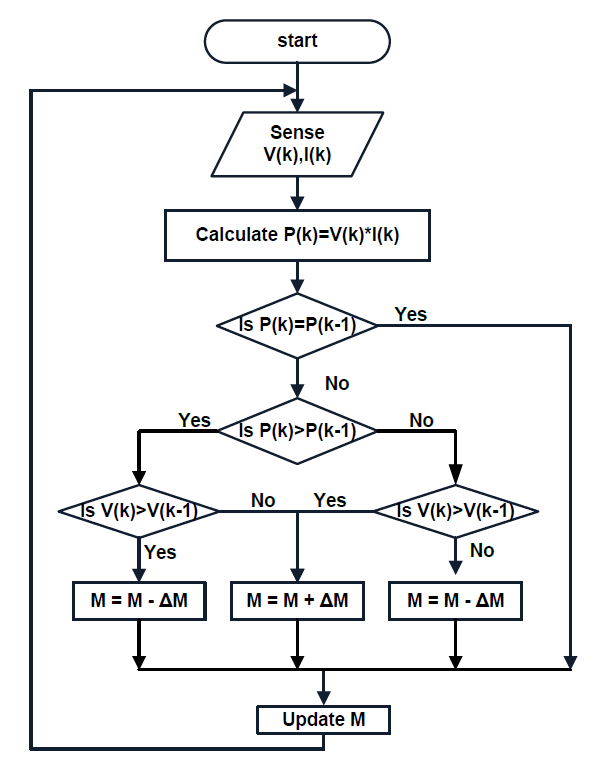
Xét tất cả giá trị X[n] có thể nhận, thử cho X[n] nhận lần lượt giá trị đó.

Thông báo cấu hình tìm được.

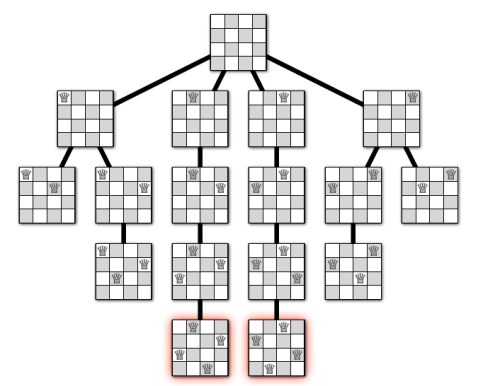
# Phần 3. Phân tích, thiết kế giải pháp

## **3.1.Lưu đồ khối của thuật toán**

3.1.1 Lưu đồ khối của thuật toán Hill ClimBing



**3.1.2 Lưu đồ khối của thuật toán BackTracking**



**3.2.Các thuật toán chính:**

**3.2.1 Code thuật toán Hill Climbing**

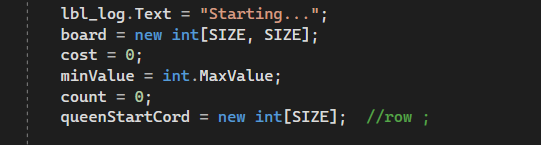
Cấu trúc dữ liệu được sử dụng cho thuật toán là Mảng một chiều :

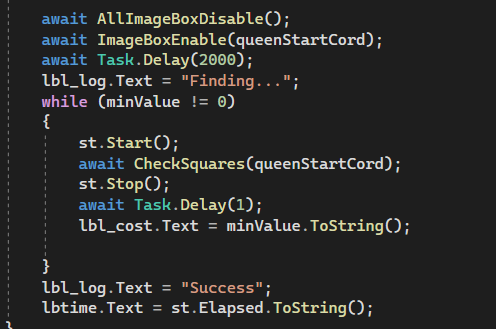
queenStartCord[]

Với queenStartCord[i]=j thì i là cột còn j là dòng ( bàn cờ nhập vào một cột chỉ được nhập 1 quân hậu )

Ngoài ra còn có mảng 2 chiều và list

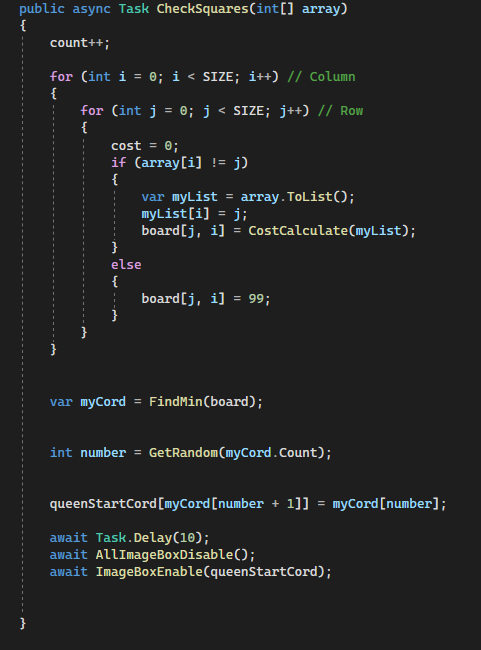
Thuật toán chính giải quyết Hill Climbing

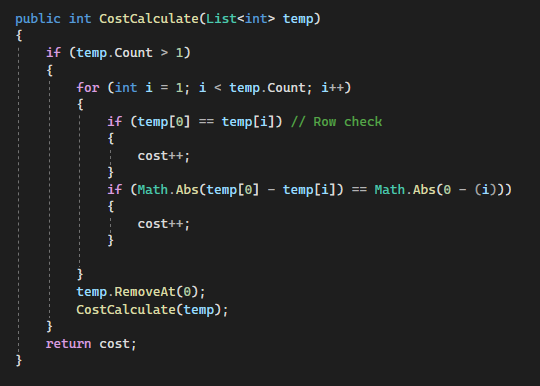




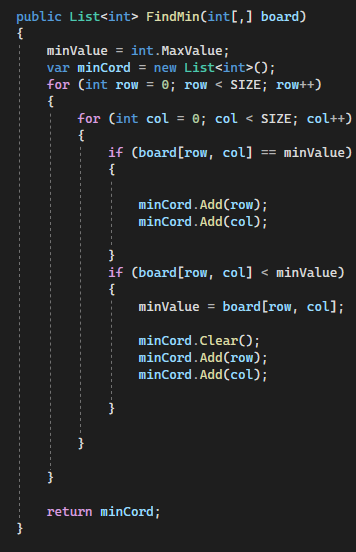
Hàm tìm kiếm trạng thái tốt hơn và thông báo cấu hình tìm được ra màn hình

Kiểm tra các vị trí không được trùng nhau ( theo hàng ngang \_vì ban đầu bàn cờ mỗi cột chỉ được nhập 1 quân hậu )

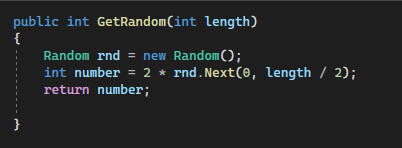


Hàm heuristic

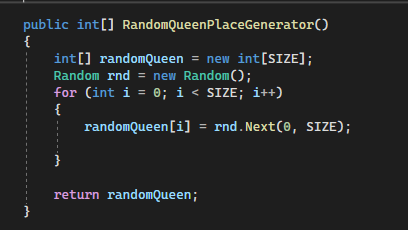
Hàm tìm kiếm trạng thái nhỏ nhất



Hàm trả về giá trị ngẫu nhiên của trạng thái nếu 2 giá trị bằng nhau



Hàm tạo ra các giá trị ngẫu nhiên của bàn cờ khi chạy với chế độ random

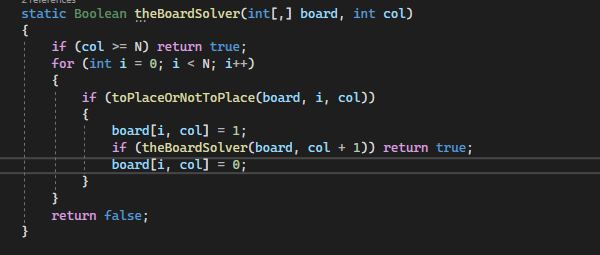


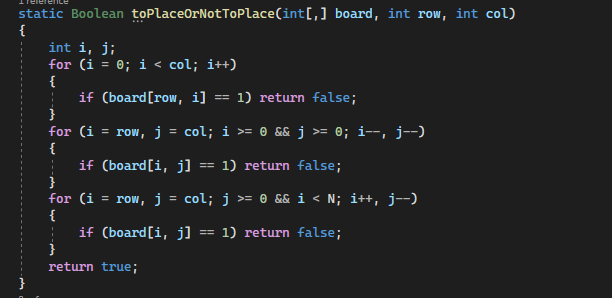
Hàm nhập trạng thái vào bàn cờ với chế độ tùy chọn



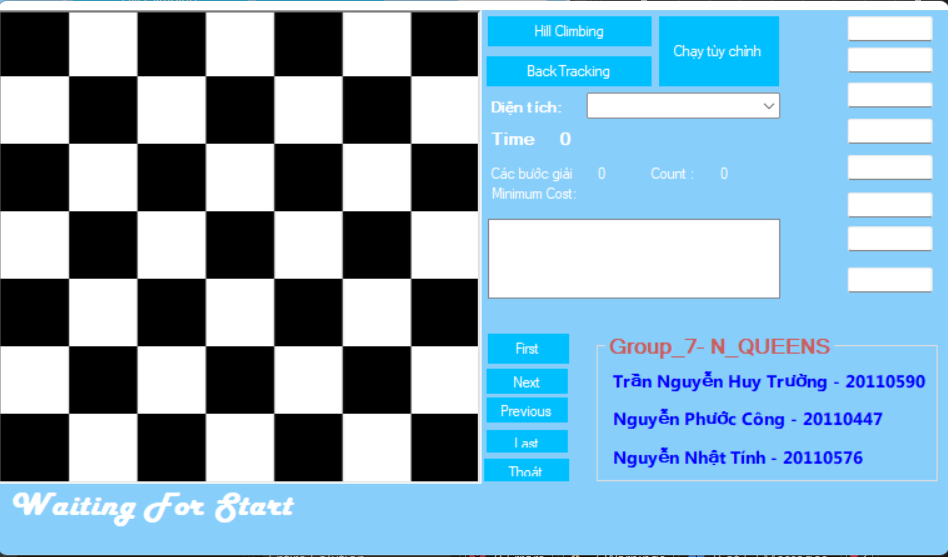
**3.2.1 Code thuật toán Backtracking**

Thuật toán chính

****

****

# Phần 4. Thực nghiệm, đánh giá, phân tích kết quả



## **4.1.Chạy thuật toán Hill Climbing**

Với thuật toán Hill Climbing ta có 2 chế độ là ramdom và tùy trọn vị trí

### **4.1.1Chạy thuật toán với Random**

Ta có thể chọn kích thước bàn cờ từ N=4 -> N=8

Với N=4





Với N=8





### **4.1.2Chạy thuật toán với tùy chọn vị trí**

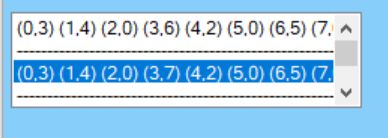
Ta chọn kích thức bàn cờ và nhập vị trí quân cờ ơ text box và nhấn nút Chọn tùy chỉnh

Với N=5 , nhập các ô theo thứ tự từ trên xuống dưới tương ứng số lượng với diện tích đã chọn ( giá trị của mỗi ô theo thứ tự là vị trí hàng của con hậu tương ứng với từng cột )





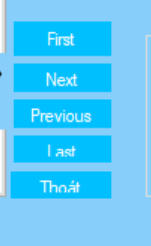
Bảng listbox thể hiện các trạng thái mà thuật toán đã giải quyết



Ta có thể click vào listbox để hiển thị lên các trạng thái của bàn cờ



Hoặc có thể sử dụng các button bên dưới để truy cập đến các trạng thái tiếp theo , trước đó , đầu tiên hoặc cuối cùng



**4.2 Chạy với thuật toán backtracking**





# Phần 5. Phần kết luận

## **5.1.Kết quả đạt được**

Chương trình đáp ứng được yêu cầu của đề tài đặt ra. Có thể áp dụng cho N quân hậu trên bàn cờ nxn (n ≥ 4).

Chương trình có giao diện đơn giản, dễ sử dụng.

Việc so sánh giữa 2 thuật toán cho thấy backtracking có nhiều ưu điểm hơn trong việc giải quyết bài toán N con hậu , Hill Climbing có nhiều trường hợp vẫn có thời gian chạy tốt

## **5.2.Hạn chế**

Nhóm còn 1 số thiếu sót trong việc hoàn thành chương trình giao diện

## **5.3.Hướng phát triển**

Chương trình mới chỉ được cài đặt trên nền tảng ngôn ngữ C#, trong tương lai sẽ phát triển trên nhiều nền tảng ngôn ngữ khác nhau ; hoàn thiện chương trình giao diện tốt hơn và thân thiện với người sử dụng

# Tài liệu tham khảo

<https://en.wikipedia.org/wiki/Hill_climbing>

<https://www.geeksforgeeks.org/introduction-hill-climbing-artificial-intelligence/>