

MÔN : LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Bài thực hành số 3.2 : Viết chương trình sắp 8 con hậu

I. Mục tiêu :

- Giúp SV ôn lại qui trình viết 1 ứng dụng VC# đơn giản, tập trung vào việc sử dụng các kiểu dữ liệu cơ bản định sẵn như struct và array.

II. Nội dung :

- Xây dựng chương trình nhỏ chạy ở chế độ textmode (console), thực hiện các yêu cầu sau :
 1. tìm tất cả phương án để sắp 8 con hậu vào bàn cờ 8x8 ô sao cho chúng không thể ăn được nhau (thỏa mãn đồng thời các điều kiện chúng không trên cùng hàng, cùng cột hay cùng đường chéo).
 2. in các phương án dạng ma trận 2 chiều để người dùng xem dễ dàng.

III. Chuẩn đầu ra :

- Sinh viên nắm vững và viết thành thạo các ứng dụng nhỏ chạy ở chế độ textmode và biết cách dùng các kiểu dữ liệu cơ bản định sẵn, nhất là kiểu struct và array, khi cần.

IV. Phân tích :

1. Ta dùng thuật giải backtracking để tìm các phương án sắp 8 con hậu như sau :

+ Lập tìm vị trí con hậu ở từng cột, từ trái sang phải :

- Nếu sắp được cho cột c thì tăng c 1 đơn vị để tiếp tục cột kế tiếp, nếu cột c đã vượt qua MAX thì đã tìm được 1 cách mới.
- Nếu không sắp được cho cột c thì giảm c 1 đơn vị để thử tiếp các khả năng còn lại của cột cũ này. Nếu c < 0 thì hết cách.

2. Để lưu vết xử lý trên từng cột, ta dùng biến array hang[MAX], mỗi phần tử hang[c] chứa chỉ số hàng đã xử lý lần cuối trước đây (-1 : chưa, MAX : hết).

Từ thuật giải được viết bằng ngôn ngữ tự nhiên ở trên, ta dịch nó thành chương trình được viết bằng VC# như trong mã nguồn dưới đây.

V. Qui trình :

1. Chạy VS .Net, chọn menu File.New.Project để hiển thị cửa sổ New Project.
2. Mở rộng mục Visual C# trong TreeView "Project Types", chọn mục Windows, chọn icon "Console Application" trong listbox "Templates" bên phải, thiết lập thư mục chứa Project trong listbox "Location", nhập tên Project vào textbox "Name:" (td. Sap8Conhau), click button OK để tạo Project theo các thông số đã khai báo.
3. Ngay sau khi Project vừa được tạo ra, cửa sổ soạn code cho chương trình được hiển thị. Hiệu chỉnh code của file Program.cs để có nội dung như sau :

```
using System;
```

```
using System.Collections.Generic;
```

```
using System.Linq;
```

```
using System.Text;
```

```
namespace Sap8Conhau
```

```
{
```

```
    class Program
```

```
    {
```

```
        //định nghĩa các thuộc tính dữ liệu
```

```
        const int MAX = 7; //chỉ số max của hàng và cột
```

```
        static int cachso = 0; //số cách sắp đã tìm được
```

```
static int[] hang = new int[MAX+1]; //danh sách chỉ số hàng đã thử trong từng cột
static int c; //vị trí cột đang sắp
```

```
//Điểm nhập của chương trình
static void Main(string[] args)
{
    // khởi động trạng thái xuất phát
    for (c = 0; c <= MAX; c++) hang[c] = -1;
    cachso = 0;
    //bắt đầu đặt con hậu ở cột 0 vào hàng 0
    hang[0] = 0;
    //bắt đầu tìm hàng cho con hậu ở cột thứ 2 (chỉ số là 1)
    c = 1;
    while (Tim1cach())
    {
        // tìm được cách sắp 8 con hậu
        InKetqua();
        //lùi lại cột trước để tiếp tục tìm cách sắp khác
        c--;
    }
    // hết cách --> dừng chương trình
    Console.WriteLine("Số nghiệm là " + cachso);
    //chờ người dùng ấn Enter để đóng cửa sổ Console lại.
    Console.Write("Ấn Enter để dừng chương trình : ");
    Console.Read();
    return;
}
```

```
//cố gắng tìm 1 phương án sắp 8 con hậu
static bool Tim1cach()
{
    int h;
    while (c <= MAX)
    {
        // tìm chỉ số hàng cho con hậu ở cột c
        h = timhang(c);
        if (h >= 0)
        { //nếu tìm được
            //lưu hàng vào danh sách kết quả rồi tăng chỉ số cột để tiếp tục
            hang[c++] = h;
            continue;
        }
        //trường hợp không tìm được hàng cho con hậu ở cột c
        if (c > 0)
        {
            //nếu cột c không phải là cột đầu tiên thì :
            //xét lại cột c từ hàng đầu rồi lùi cột c
            hang[c--] = -1;
            continue;
        }
        //hết cách
        return false;
    }
}
```

```

    }
    return true;
} //hết hàm Tim1cach

//hàm cố gắng tìm chỉ số hàng cho con hậu ở cột c
//trả về chỉ số hàng tìm được hay -1 để báo sai
static int timhang(int c)
{
    int h, hmin = hang[c] + 1;
    for (h = hmin; h <= MAX; h++) // thử hàng h
        if (testvitri(h, c)) return h;
    return -1;
}

//hàm kiểm tra xem có thể đặt con hậu ở vị trí h,c ?
//trả về TRUE nếu được, FALSE nếu không
static bool testvitri(int h, int c) {
    int c1, h1;
    //cột c đang kiểm tra nên chưa có con hậu nào
    // xem có con hậu nào nằm ở hàng h ?
    for (c1 = 0; c1 < c; c1++)
        if (hang[c1] == h) return false;
    // xem có con hậu nào nằm trên đường chéo trên trái - dưới phải
    c1 = c-1; h1 = h-1;
    while (c1 >= 0 && h1 >= 0) {
        if (hang[c1] == h1) return false;
        h1--; c1--;
    }
    // xem có con hậu nào nằm trên đường chéo trên phải - dưới trái
    c1 = c-1; h1 = h+1;
    while (c1 >= 0 && h1 <= MAX) {
        if (hang[c1] == h1) return false;
        h1++; c1--;
    }
    return true;
}

//hàm in kết quả 1 cách sắp 8 con hậu theo yêu cầu
static void InKetqua() {
    int h, c;
    Console.WriteLine("Cách sắp thứ : " + (++cachso));
    for (h = 0; h <= MAX; h++) {
        // Hiển thị hàng lưới ngang của bàn cờ
        for (c = 0; c <= MAX; c++) Console.Write("+---");
        Console.WriteLine("+");
        // Hiển thị nội dung hàng thứ h của bàn cờ
        for (c = 0; c <= MAX; c++)
            if (hang[c] == h) Console.Write("| x ");
            else Console.Write("| ");
        Console.WriteLine("|");
    }
    // Hiển thị hàng lưới ngang cuối cùng của bàn cờ

```

```
        for (c = 0; c <= MAX; c++) Console.Write("+---");  
        Console.WriteLine("+");  
    }  
  
    }//hết class Program  
}//hết namespace
```

4. Chọn menu Debug.Start Debugging để dịch và chạy ứng dụng. Hãy kiểm tra kết quả hiển thị của chương trình.