

MSSV: 23520980

Họ và tên: Nguyễn Phương Nam

Lớp: CS112.P22

Bài tập Bonus

Thiết kế phần mềm giải bài toán đường đi an toàn

1. Phát biểu bài toán

- Cho một bản đồ hình vuông kích thước $n \times n$, trong đó mỗi ô có thể là:
 - **Ô trắng**: Có thể đi qua (an toàn).
 - **Ô đen**: Không thể đi qua (chướng ngại vật).
- Người dùng được phép di chuyển **chỉ sang phải hoặc xuống dưới** từ ô (0,0) đến ô (n-1,n-1).
- Yêu cầu:
 - Đếm số đường đi an toàn từ ô đầu đến ô cuối.
 - Liệt kê tất cả các đường đi đó.
 - Cho phép người dùng tạo bản đồ, đánh dấu chướng ngại, và xem trực tiếp từng đường đi.

2. Chức năng của phần mềm

- Nhập kích thước bản đồ tùy ý (tối đa 20×20).
- Tự động sinh lưới bản đồ.
- Người dùng nhấp vào các ô để đánh dấu chướng ngại vật.
- Tính toán:
 - Tổng số đường đi an toàn (Dynamic Programming).
 - Danh sách tất cả các đường đi (Backtracking).
- Hiển thị rõ ràng từng đường đi dưới dạng tọa độ.
- Cho phép xem trực tiếp từng đường đi trên bản đồ bằng màu xanh lá.
- Nút Reset để tạo lại bản đồ trắng.

3. Ý tưởng giải thuật

- Phần mềm chia bài toán thành 2 phần chính:
 - o Đếm số đường đi (chỉ đi phải hoặc xuống): dùng lập trình động (DP) để tính nhanh. (mục đích là muốn sài giải thuật dinamic programming).
 - o Liệt kê toàn bộ đường đi: dùng tìm kiếm quay lui (Backtracking) để duyệt hết các đường hợp lệ. (Muốn thêm tính năng này nên sài backtracking).
- Kết hợp với giao diện WPF, phần mềm cho phép tương tác trực quan và xem đường đi dễ dàng.

4. Chi tiết thuật toán

- Thuật toán đếm đường đi - Dynamic Programming:
 - o Gọi $dp[i][j]$ là số đường đi an toàn từ (0,0) đến (i,j).
 - Công thức: $dp[i][j] = dp[i-1][j] + dp[i][j-1]$ (nếu ô (i,j) không bị chặn)
 - Điều kiện biên:
 - $dp[0][0] = 1$ nếu ô (0,0) không bị chặn.
 - Các ô bị chặn thì $dp[i][j] = 0$.
 - Độ phức tạp: $O(n^2)$.
- Thuật toán liệt kê đường đi – Backtracking:
 - o Bắt đầu từ (0,0), tại mỗi bước, thử đi:
 - Sang phải (i, j+1) nếu hợp lệ.
 - Xuống dưới (i+1, j) nếu hợp lệ.
 - o Mỗi khi đến được ô (n-1,n-1), lưu lại đường đi hiện tại vào danh sách kết quả.
 - o Độ phức tạp: có thể lên đến $O(2^{(2n)})$ nếu không có chương ngại.

5. Cấu trúc dữ liệu sử dụng

Mục đích	Cấu trúc dữ liệu
Bản đồ các ô	Button[,] cells
Ghi nhận ô hợp lệ	bool[,] grid
Đếm đường đi	int[,] dp
Lưu danh sách các đường đi	List<List<(int,int)>>
Hiển thị đường đi	ListBox, TextBlock trong XAML

Link source trên github: <https://github.com/NamPhuong550805/SafePath>