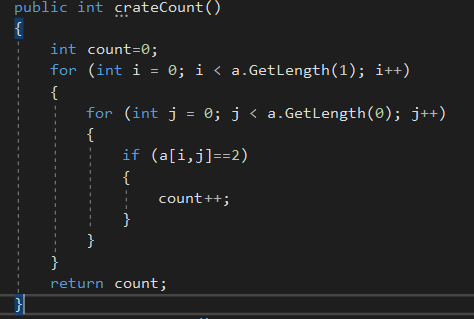
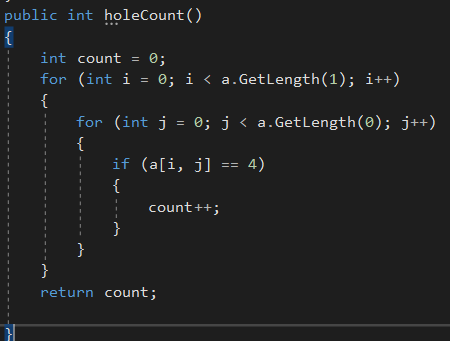
# Hàm CratesCount:



Dùng để đến bao nhiêu hộp trong 1 ma trận 10x10

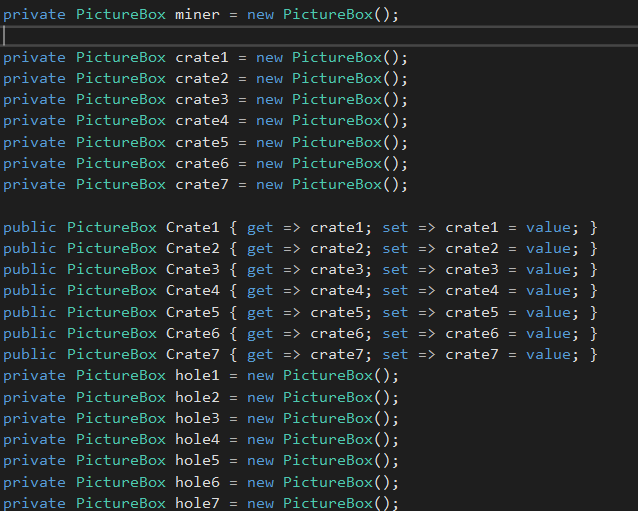
Nếu dòng I cột j ma trận là 2 thì đếm tăng lên sau đó trả về.

# Hàm holeCount

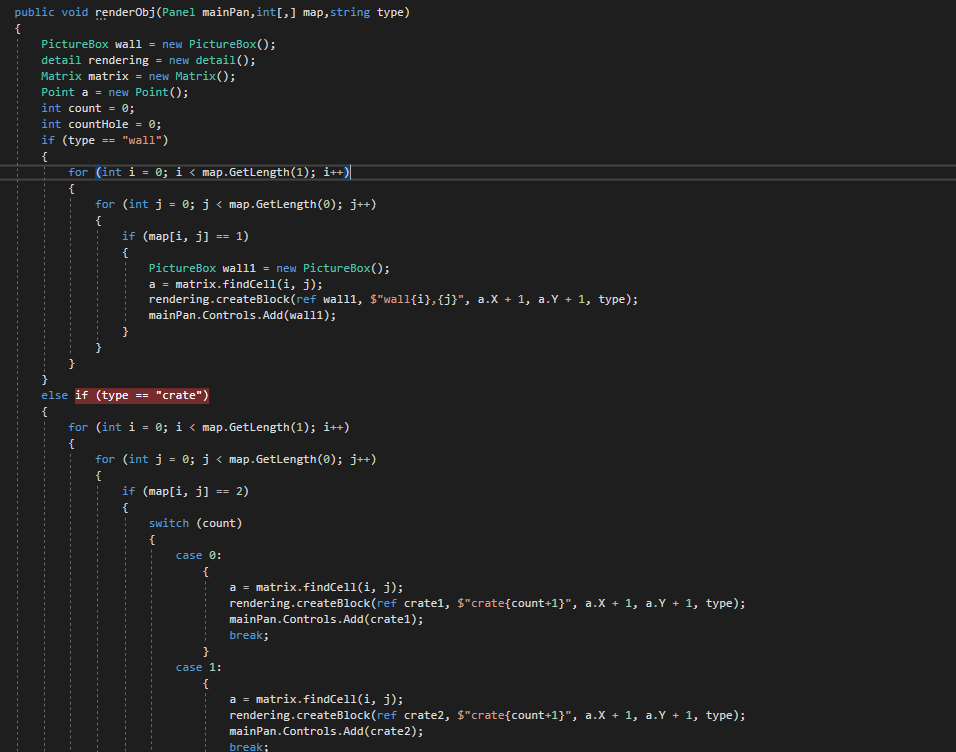


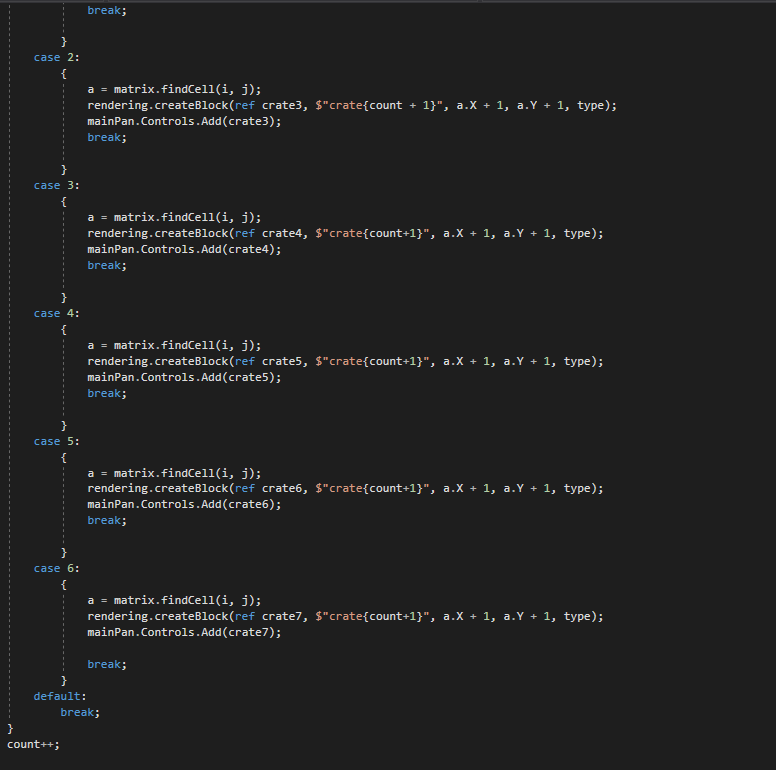
Hàm đếm hố. cũng tương tự với hàm đếm hộp. nếu là 4 thì đếm và trả về

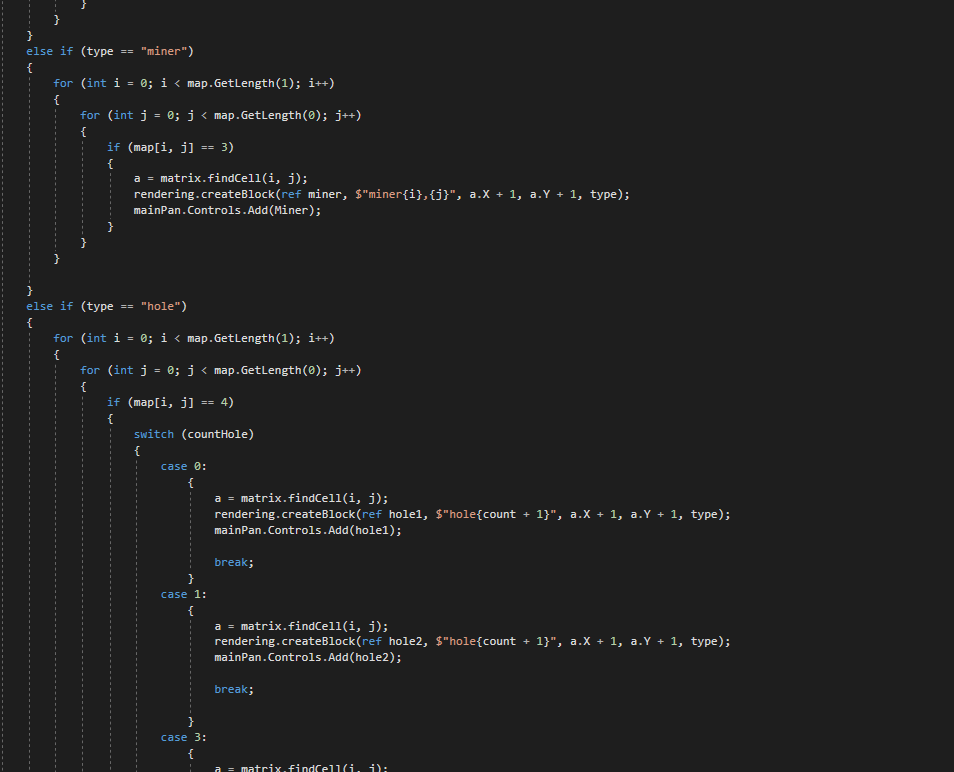
# Các pictureBox:

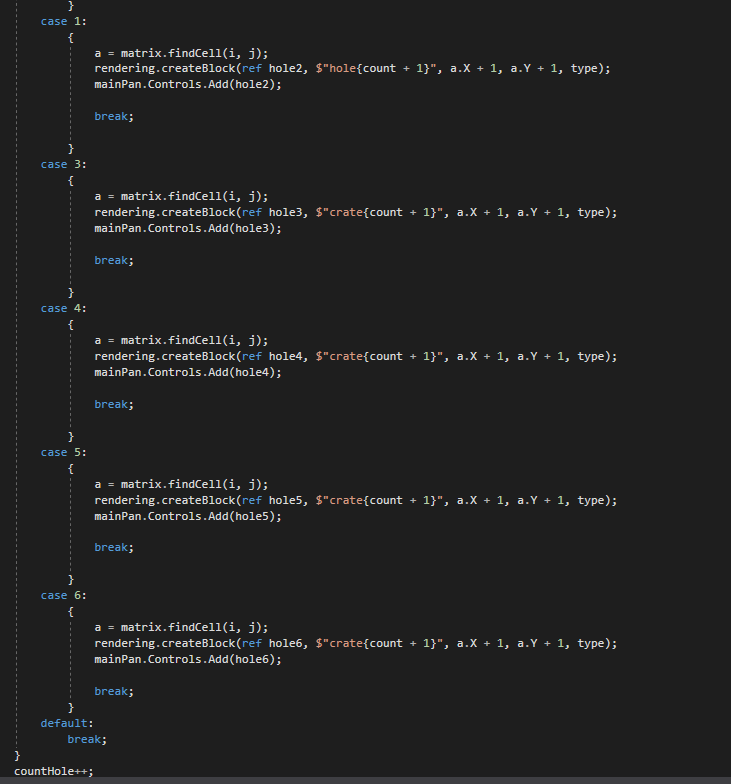


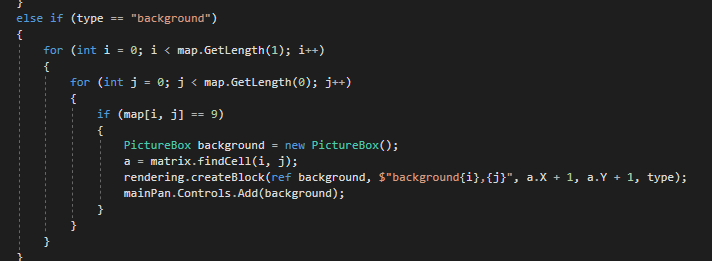
# Hàm RenderObj



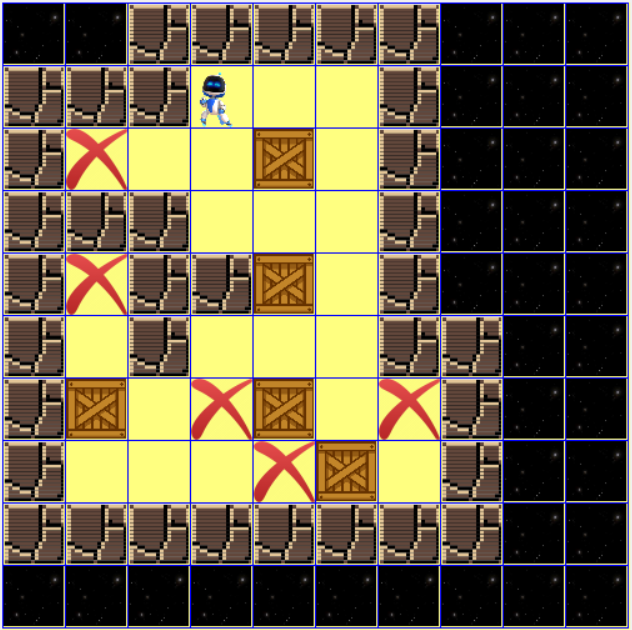








Đây là hàm quan trọng của game. Là hàm để tạo các bức tường, người đẩy hộp, hộp và hố thắng.



ở đây có chia thành 3 loại

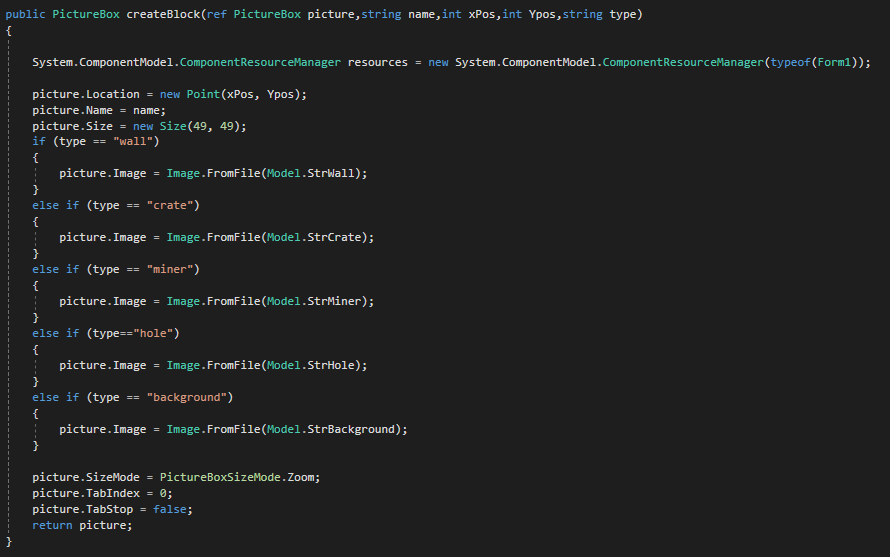
1. Là wall là những bức tường.
2. Crate là hộp
3. Hole là hố thắng.
4. Miner là người đẩy.
5. Background là hình màu đen

Tất cả được chia nên chỉ cần duyệt ma trận tương tự với 2 là crate, 4 là hole, 3 là miner,9 là background

Tất cả đều được duyệt qua ma trận 10x10 để hiển thị lên form. Thay vì dùng cách kéo thả cơ bản. cách này sẽ tiện hơn và qua màn không phải kéo thả lại.

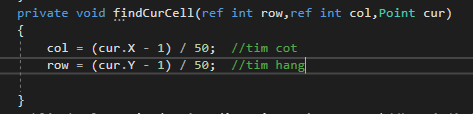
Có tối đa 6 hộp và 6 hố.

# Hàm createBlock



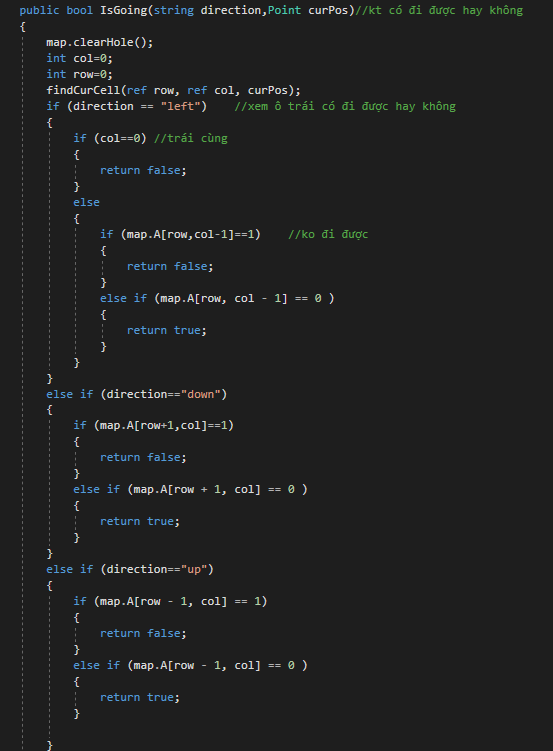
Hàm sẽ tạo đối tượng trên win dựa vào tên truyền vào, vị trí,hình ảnh. Độ lớn đều bằng nhau là 49x49px. Tất cả là picturebox.

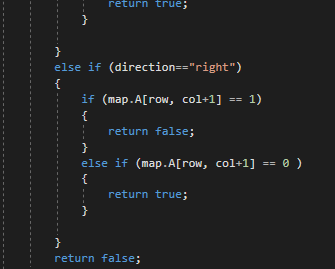
# Hàm tìm ô hiện tại



Đây là hàm tìm ô hiện tại ở ma trận 10x10. Ánh xạ từ vị trí form

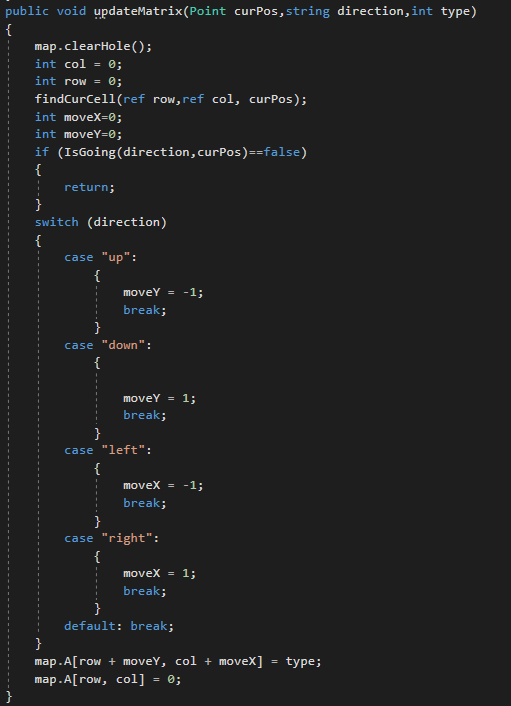
# Hàm kt miner đi được ô tiếp theo không





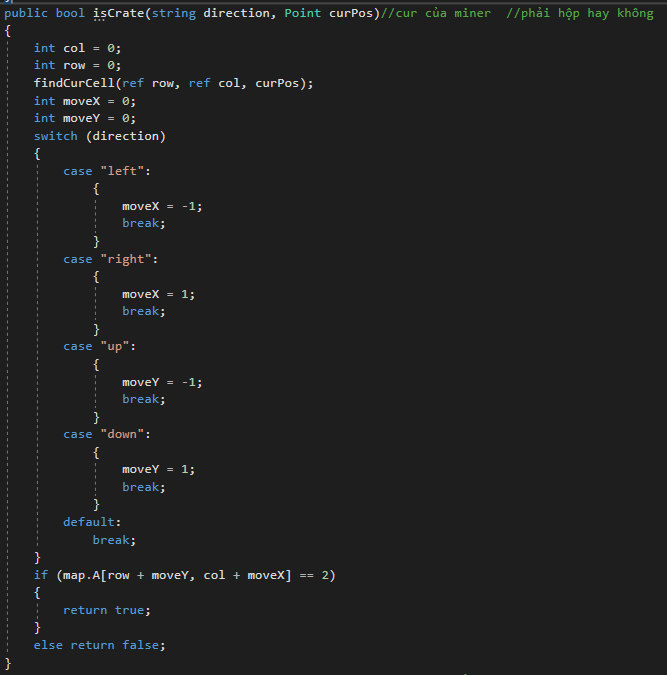
Là hàm chính trong game. Dựa vào hướng mà người chơi nhập (ví dụ: left, right, up, down) để kiểm tra xem ô tiếp đó có đi được không. Đi được là khi ô kế tiếp là 0 thì trả về true, ngược lại là false. Hàm chỉ cần nhập hướng và ô hiện tại trên bàn.

# Hàm updateMatrix



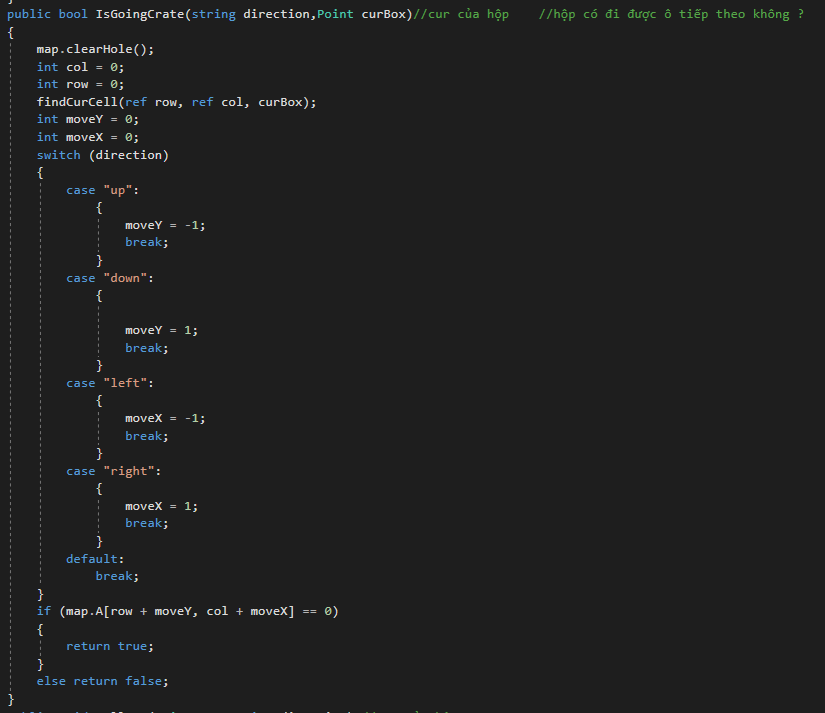
Hàm này sử dụng khi đã di chuyển 1 ô và cập nhật lại trên ma trận. điền số 0 vào ô đã rời đi và điền số loại: (ví dụ là crate hoặc miner ) vào ô tiếp theo.

# Hàm isCrate



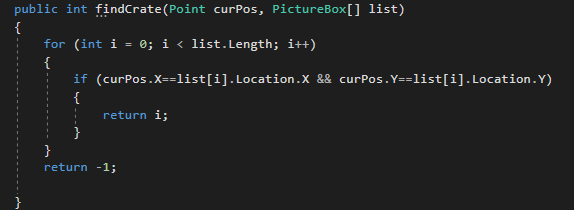
Đây là hàm kiểm tra Ô TIẾP THEO có phải là hộp hay không. Coi từ vị trí của miner khi nhập hướng và vị trí. Nếu ô tiếp theo là hộp thì trả true ngược lại false

# Hàm isgoingCrate



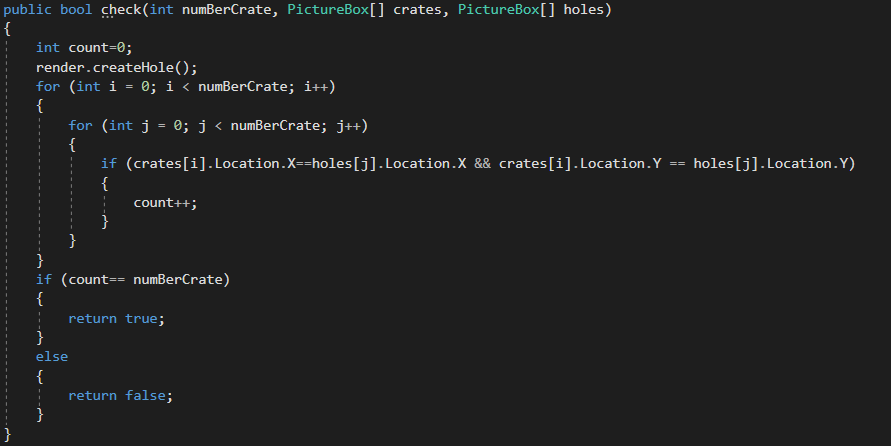
Kiểm tra xem Ô TIẾP THEO CỦA HỘP có đi được hay không. Cũng giống như kiểm tra miner đi được không. Cần hướng và vị trí của Crate. Tùy theo hướng mà hàm sẽ xác định đi được không. Đi được thì true ngược lại false

# Hàm tìm hộp



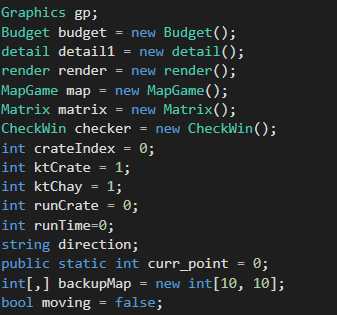
Tìm hộp trên ma trận form. Là hàm bool

# Hàm CheckWin

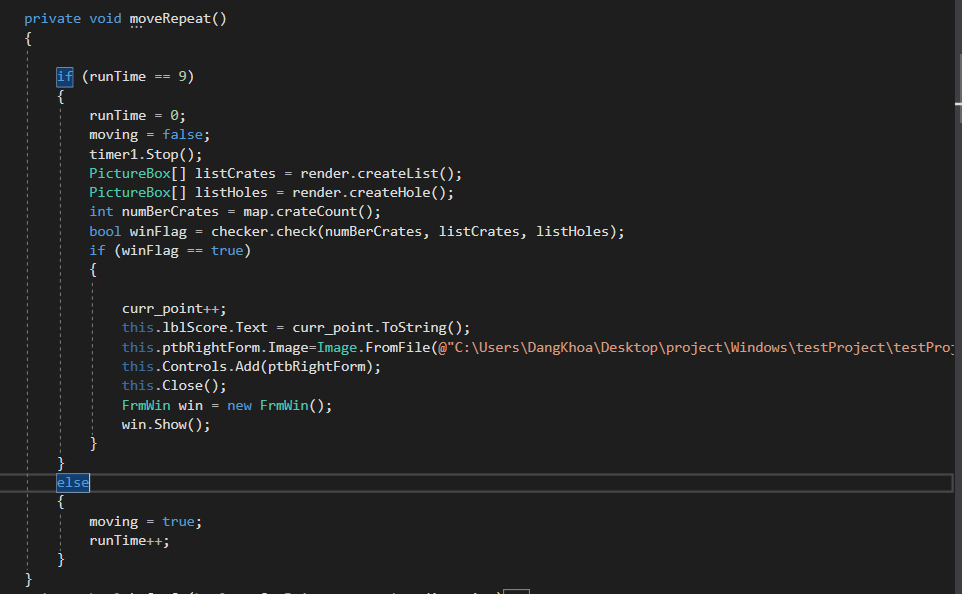


Là hàm kiểm tra điều kiện thắng khi tất cả hộp nằm đúng vào tất cả các hố. nếu số hộp đủ với số hố đã được tạo thì thắng.

# Các biến toàn cục và cờ

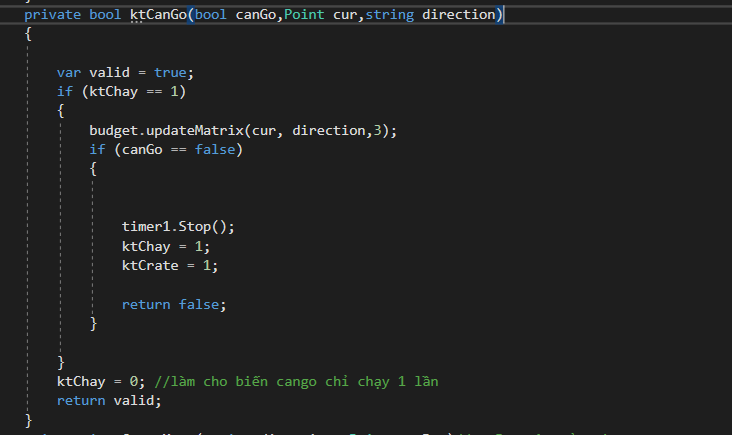


# Hàm repeat:

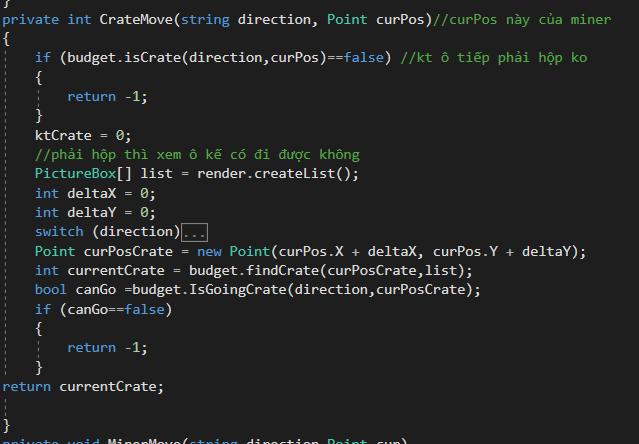


Dùng để lập lại để di chuyển đối tượng từ ô này qua ô khác

# Hàm cango:

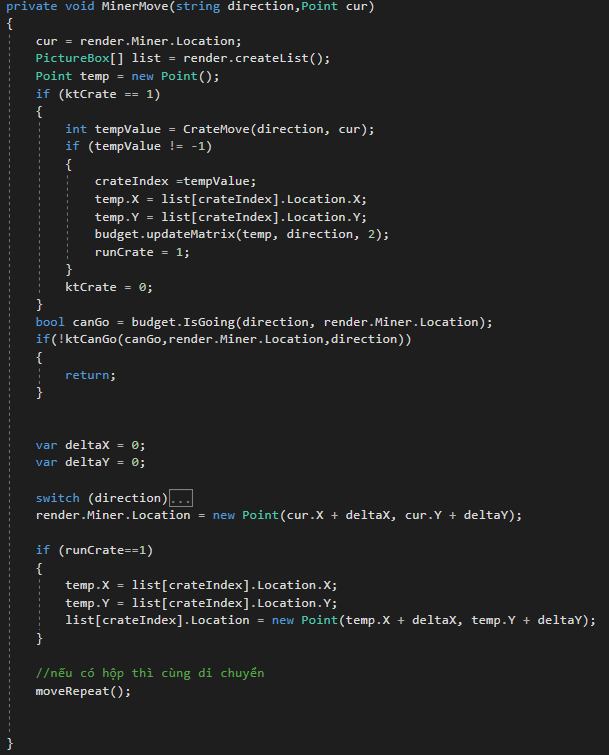


Kiểm tra chỉ 1 lần xem đi được không



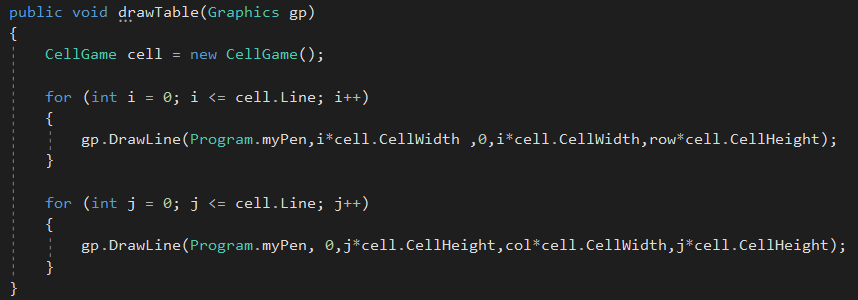
Hàm trả về vị trí hộp có thể đi được.

# Hàm MinerMover



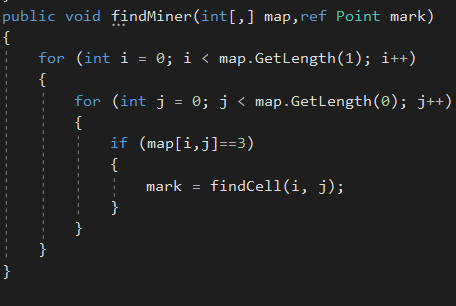
Khi đã xác định đi được thì hàm này sẽ chạy. và nếu kèm hộp đi được thì 2 vật cùng di chuyển sang 2 ô tiếp theo.

# Hàm fincell



Tìm ô từ form

# Hàm findMiner



Hàm tìm vị trí miner