# Trạng thái bài toán

Sudoku\_GA sẽ chạy qua nhiều thế hệ (generations). Một thế hệ chứa một quần thể (population) của các bảng sudoku không trống. Các bảng trong quần thể sẽ được sắp xếp theo điểm fitness. Điểm fitness của mỗi cá thể sudoku được đánh giá bằng số lần hai giá trị lặp lại trong một bảng.

## Trạng thái khởi đầu

Thế hệ đầu tiên chứa quần thể tạo ra từ bảng cho trước.

## Trạng thái kết thúc

Thế hệ chứa quần thể chứa ít nhất một cá thể có điểm fitness bằng 0.

# Luật di chuyển

Tổng quan:

Ở mỗi thế hệ, quần thể sẽ sắp xếp các bảng theo thứ tự điểm fitness từ thấp đến cao. Các cá thể có điểm fitness thấp nhất và một vài cá thể bất kỳ sẽ được chọn để phối giống và tạo ra thế hệ sau. Sau đó mỗi bảng con sẽ được chọn ngẫu nhiên để gây đột biết (mutate). Cuối cùng, các bảng con trở thành thể mới cho thế hệ sau.

## Creating children from breeders

Quy tắc để tạo ra bảng con: Chọn hai cá thể phối giống (breeders) như trên. Lấy các grid của mỗi cá thể và shuffle chúng. Sau đó, chọn ra đủ số grid đầu tiên để tạo thành bảng mới. Lặp lại quá trình cho đến tạo đủ số con bằng quần thể ban đầu. Ví dụ: ở bảng 9x9, lấy 9 grid của cha và 9 grid của mẹ, sau đó chọn 9 grid bất kỳ từ 18 grid trên.

### Mutation

mutation\_rate sẽ quyết định tỉ lệ của quần thể bị “biến đổi”. Quá trình mutation hoạt động như sau: trong một đứa con bị mutated, chọn một ô lớn bất kỳ và hoán đổi giá trị 2 ô KHÔNG ĐƯỢC ĐỊNH TRƯỚC (non-fixed). Mutation giúp cho quần thể có tính đa dạng, tạo khả năng thoát khỏi điểm cực tiểu cục bộ.

Tuy nhiên, giá trị fit vẫn có thể không cải thiện khi điểm fitness tiến gần tới 0, bởi vì quần thể rất khó tạo ra đứa con có điểm fit tối ưu hơn. Vì thế, qua mỗi thế hệ mà điểm fitness tốt nhất không được cải thiện, mutation\_rate sẽ được tăng thêm. Hàm biểu diễn mutation rate là:

Total rate = Rate + Bonus rate

Trong đó Bonus rate được tính bằng:

Bonus rate = (0.85 - Rate) \* Generations without improvement / Max generations

Nếu giải thuật có khởi động lại sau vài thế hệ (xem ở phía dưới), thì sẽ lấy số thế hệ trước khi reset thay vì số thế hệ toàn cục. Ở hàm trên, mutation rate sẽ không bao giờ vượt quá 0.85 nhằm đảm bảo một phần quần thẻ giữ tính ổn định của gen, tránh điểm fitness không còn hội tụ về 0.

### Restart after n generations wihtout improvement

Sau n thế hệ mà điểm fitness không cải thiện, chương trình sẽ tái khởi động: Tạo một quần thể mới và chạy chương trình phối giống trên quần thể đó. Điều này có thể giúp quần thể thoát khỏi tình trạng ổn định về gen, dẫn đến không cải thiện về điểm fitness. Giải thuật sẽ không restart vô thời hạn: Tổng số thế hệ đã trải qua của chương trình không được lớn hơn giá trị tối đa được định sẵn từ trước.

Mỗi khi tái khởi động, giải thuật sẽ chọn đứa con có điểm fitness **tệ nhất** (và **tốt nhất** nếu điểm fitness của nó đủ tốt) của thế hệ cuối cùng và bỏ vào quần thể mới. Điều này giúp thực hiện 2 điều: 1) Giữ lại gen tốt nhất từ quần thể trước và 2) Tạo tính đa dạng cho gen.

Nguyên nhân tại sao ưu tiên điểm fitness **tệ nhất** thay vì **tốt nhất** là nhằm tránh tình trạng rơi lại vào vùng cực tiểu cục bộ do bảo toàn về gen của quần thể. Đứa con có điểm fitness **tốt** **nhất** được chọn giữ lại cho quần thể mới chỉ khi nó có điểm fitness đủ thấp (được quy định bởi biến best\_score\_to\_keep)