

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



HƯỚNG DẪN CHẠY
Ứng dụng Sudoku JS

Giảng viên hướng dẫn: TS. Nguyễn Tuấn Dũng

Nhóm sinh viên thực hiện

Phạm Hữu Bảo Chung	20140479
Roãn Văn Thụ	20144416
Trần Trung Hiếu	20141568
Bùi Như Minh	20142887

Hà Nội, 2018

MỤC LỤC

PHẦN 1. MÔ TẢ BÀI TOÁN	2
1. Giới thiệu bài toán.....	2
1.1. Đầu vào.....	3
1.2. Đầu ra.....	3
1.3. Phác thảo thuật toán.....	4
2. Bài toán Sudoku trong ứng dụng Sudoku JS.....	4
2.1. Sinh cấu hình ban đầu hợp lệ.....	4
2.2. Kiểm tra tính hợp lệ của cấu hình.....	5
PHẦN 2. ỨNG DỤNG SUDOKU JS	6
1. Hướng dẫn chạy mã nguồn	6
2. Hướng dẫn sử dụng ứng dụng.....	6

PHẦN 1. MÔ TẢ BÀI TOÁN

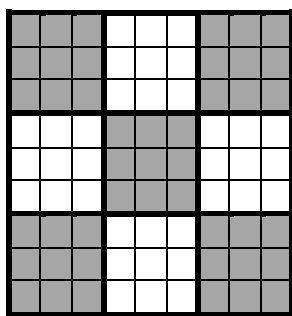
1. Giới thiệu bài toán

Trò chơi Sudoku được tiến hành trên một lưới vuông kích thước 9×9 . Mỗi ô trên lưới có thể nhận một trong 9 giá trị $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Mỗi phép gán một số ô trên lưới những giá trị nhất định cho ta một cấu hình. Cấu hình có thể là đầy đủ (tất cả các ô trên lưới đều được gán giá trị) hoặc không đầy đủ (một số ô trên lưới chưa được gán giá trị).

Một cấu hình được gọi là hợp lệ nếu nó là một cấu hình đầy đủ thỏa mãn các điều kiện dưới đây:

1. Hai ô bất kỳ trong cùng một hàng luôn được gán giá trị khác nhau.
2. Hai ô bất kỳ trong cùng một cột luôn được gán giá trị khác nhau.
3. Hai ô bất kỳ trong cùng một lưới 3×3 luôn được gán giá trị khác nhau.



Hình 1-1. Các lưới 3×3 trên lưới Sudoku.

Trong bài toán Sudoku, từ một cấu hình ban đầu, ta tìm cách gán giá trị cho các ô trống trong cấu hình này để thu được một cấu hình hợp lệ.

			2					
						3		8
5		3	1	9				6
			9					
4	2			3				
	3			2	6	1		
		9					2	
2	4		7			8		
3		7		1				

Hình 1-2. Cấu hình ban đầu.

6	8	4	2	7	3	5	9	1
1	9	2	6	4	5	3	7	8
5	7	3	1	9	8	2	4	6
7	6	5	9	8	1	4	3	2
4	2	1	5	3	7	6	8	9
9	3	8	4	2	6	1	5	7
8	1	9	3	6	4	7	2	5
2	4	6	7	5	9	8	1	3
3	5	7	8	1	2	9	6	4

Hình 1-3. Cấu hình hợp lệ.

1.1. Đầu vào

Đầu vào của bài toán Sudoku truyền thống là:

- Một mảng hai chiều T_0 kích thước 9×9 mô tả cấu hình ban đầu của bảng Sudoku.
- Giá trị 0 được sử dụng để thay thế cho các ô trống trên lưới.

Ví dụ, đầu vào của bài toán Sudoku được mô tả ở hình (1.2) là:

$$T_0 = \begin{bmatrix} [0, 0, 0, 2, 0, 0, 0, 0, 0], \\ [0, 0, 0, 0, 0, 0, 3, 0, 8], \\ [5, 0, 3, 1, 9, 0, 0, 0, 6], \\ [0, 0, 0, 9, 0, 0, 0, 0, 0], \\ [4, 2, 0, 0, 3, 0, 0, 0, 0], \\ [0, 3, 0, 0, 2, 6, 1, 0, 0], \\ [0, 0, 9, 0, 0, 0, 0, 2, 0], \\ [2, 4, 0, 7, 0, 0, 8, 0, 0], \\ [3, 0, 7, 0, 1, 0, 0, 0, 0] \end{bmatrix}$$

1.2. Đầu ra

Đầu ra của bài toán Sudoku truyền thống là:

- Mảng hai chiều T_{sol} kích thước 9×9 mô tả một cấu hình hợp lệ xuất phát từ cấu hình ban đầu, hoặc
- Kết luận cấu hình ban đầu không có phương án giải.

Ví dụ, đầu ra của bài toán Sudoku được mô tả ở hình (1.3) là:

$$T_{sol} = \begin{bmatrix} [6, 8, 4, 2, 7, 3, 5, 9, 1], \\ [1, 9, 2, 6, 4, 5, 3, 7, 8], \\ [5, 7, 3, 1, 9, 8, 2, 4, 6], \\ [7, 6, 5, 9, 8, 1, 4, 3, 2], \\ [4, 2, 1, 5, 3, 7, 6, 8, 9], \\ [9, 3, 8, 4, 2, 6, 1, 5, 7], \\ [8, 1, 9, 3, 6, 4, 7, 2, 5], \\ [2, 4, 6, 7, 5, 9, 8, 1, 3], \\ [3, 5, 7, 8, 1, 2, 9, 6, 4] \end{bmatrix}$$

1.3. Phác thảo thuật toán

Bài toán Sudoku truyền thống có thể được giải theo phương pháp quay lui. Theo đó, ở mỗi bước, ta sẽ chọn ô trống có số lượng giá trị hợp lệ nhỏ nhất trên lưới làm ứng cử viên. Với mỗi giá trị của ứng cử viên, ta tiến hành gọi đệ quy và thực hiện bước kế tiếp.

Nếu tất cả các ô trống trên lưới đều được duyệt qua, tức là, có một phép gán giá trị cho chúng để thu được một cấu hình hợp lệ, ta trả về cấu hình này và kết thúc thuật toán.

Ngược lại, nếu số lượng giá trị hợp lệ của ứng cử viên ở một bước nào đấy bằng 0, ta sẽ tiến hành quay lui, gán một giá trị hợp lệ khác cho ứng cử viên ở bước trước đó rồi tiếp tục thuật toán.

Lưu ý, để xác định một giá trị $v \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ có phải là giá trị hợp lệ của $T[x][y]$ hay không, ta cần kiểm tra các điều kiện:

- $v \neq T[x][j], \quad j = \overline{0, 8}$
- $v \neq T[i][y], \quad i = \overline{0, 8}$
- $v \neq T[i][j], \quad i = (x - x \% 3) + \lfloor k/3 \rfloor$
 $j = (y - y \% 3) + (k \% 3), k = \overline{0, 8}$

2. Bài toán Sudoku trong ứng dụng Sudoku JS

Ứng dụng Suduko JS được xây dựng nhằm mô phỏng trò chơi Sudoku truyền thống. Nghĩa là, với một cấu hình Sudoku ban đầu, người chơi được phép điền giá trị vào các ô trống trên lưới Sudoku. Sau khi tất cả các ô đã được điền, ứng dụng cần kiểm tra tính hợp lệ của cấu hình và thông báo kết quả về phía người dùng.

Vì ứng dụng cho phép người chơi tham gia vào quá trình giải cấu hình ban đầu của trò chơi nên chức năng giải cấu hình Sudoku trở thành thứ yếu. Thay vào đó, chương trình ứng dụng cần có khả năng sinh ra được một cấu hình ban đầu hợp lệ và có khả năng kiểm tra tính hợp lệ của cấu hình mà người dùng tạo ra.

2.1. Sinh cấu hình ban đầu hợp lệ

Một cấu hình ban đầu được gọi là hợp lệ nếu từ cấu hình này, ta chỉ sinh ra được đúng một cấu hình hợp lệ. Như vậy, bài toán sinh cấu hình ban đầu hợp lệ có:

- **Đầu vào** là một mảng hai chiều T kích thước 9×9 . Trong đó, tất cả các phần tử đều có giá trị bằng 0.
- **Đầu ra** là mảng hai chiều T_0 kích thước 9×9 mô tả một cấu hình ban đầu hợp lệ.

Ta có thể giải bài toán này thông qua giải bài toán Sudoku truyền thống với cấu hình ban đầu bằng rỗng. Sau khi giải bài toán Sudoku truyền thống và tìm được một cấu hình hợp lệ, ta tìm cách loại bỏ một số phần tử trong cấu hình này để thu được một cấu hình ban đầu hợp lệ.

2.2. Kiểm tra tính hợp lệ của cấu hình

Bài toán kiểm tra tính hợp lệ của cấu hình có:

- **Đầu vào** là một mảng hai chiều T_{sol} kích thước 9×9 mô tả một cấu hình đầy đủ (cấu hình này do người dùng tạo ra trong quá trình chơi).
- **Đầu ra** là một giá trị boolean xác định tính hợp lệ của T_{sol} .

Bài toán này có thể được giải một cách dễ dàng thông qua bộ phận kiểm tra tính hợp lệ của bài toán Sudoku truyền thống.

PHẦN 2. ỨNG DỤNG SUDOKU JS

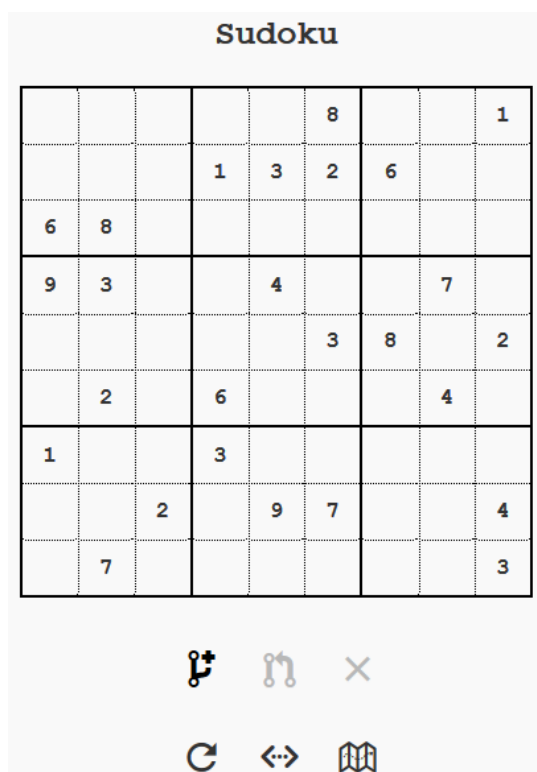
1. Hướng dẫn chạy mã nguồn

Thư mục **sudoku** chứa hệ thống các thư mục và tệp tin phục vụ xây dựng chương trình ứng dụng, bao gồm:

- Tệp tin **sudoku.html** là tệp thực thi chương trình ứng dụng.
- Tệp tin **sudoku.js** là tệp chứa mã nguồn chương trình.
- Tệp tin **style.css** là tệp hỗ trợ xây dựng giao diện chương trình ứng dụng.
- Hai thư mục **fonts** và **iocnic** lưu trữ font chữ được sử dụng trong ứng dụng.
- Thư mục **images** lưu trữ các hình ảnh được sử dụng trong ứng dụng.
- Các thư viện **jquery-1.11.1.min**, **jquery.mobile-1.4.5.min**.

Để chạy chương trình ứng dụng, ta mở tệp **sudoku.html** trên một trình duyệt nào đó.

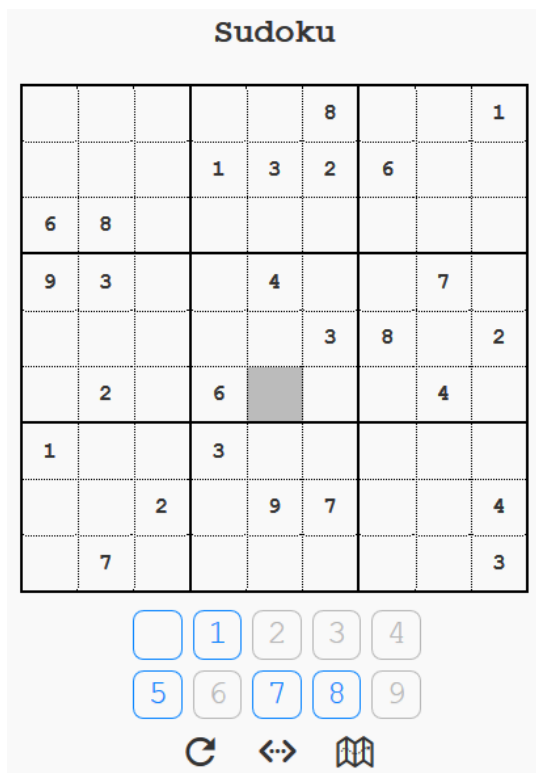
2. Hướng dẫn sử dụng ứng dụng






Hình 2-1. Giao diện ứng dụng Sudoku JS.

Giao diện ứng dụng Sudoku JS bao gồm 2 thành phần chính:


- Lưới Sudoku, được sử dụng để lưu giữ cấu hình bảng Sudoku. Người dùng có thể ấn vào một ô trống bất kỳ trên lưới để gán một giá trị hợp lệ cho ô này.







Hình 2-2. Giao diện ứng dụng sau khi chọn một ô trên lưới.

- Nhóm các phím chức năng.
 - Nhóm phím ở hàng trên cho phép chuyển từ trạng thái ổn định sang trạng thái tạm thời (phím ) và ngược lại (các phím  và ).

Ở **trạng thái ổn định**, các phép gán giá trị của người dùng sẽ được lưu trữ một cách ổn định. Khi đó, để xóa một loạt các ô, ta sẽ phải xóa lần lượt từng ô một.

Ở **trạng thái tạm thời**, các phép gán giá trị của người dùng sẽ được lưu trữ một cách tạm thời. Nghĩa là, người dùng có thể thực hiện việc đoán giá trị trên các ô. Nếu muốn lưu trữ các phép đoán này, ta ấn phím . Ngược lại, nếu muốn xóa

bỏ các phép đoán này, ta ấn phím , ứng dụng sẽ quay trở về trạng thái ổn định trước đó.

- Nhóm phím ở hàng dưới cho phép người dùng bắt đầu lại trò chơi trên lưới cũ (phím ) , bắt đầu lại trò chơi trên lưới mới (phím ) hoặc gọi đến dịch vụ hỗ trợ (phím ).