

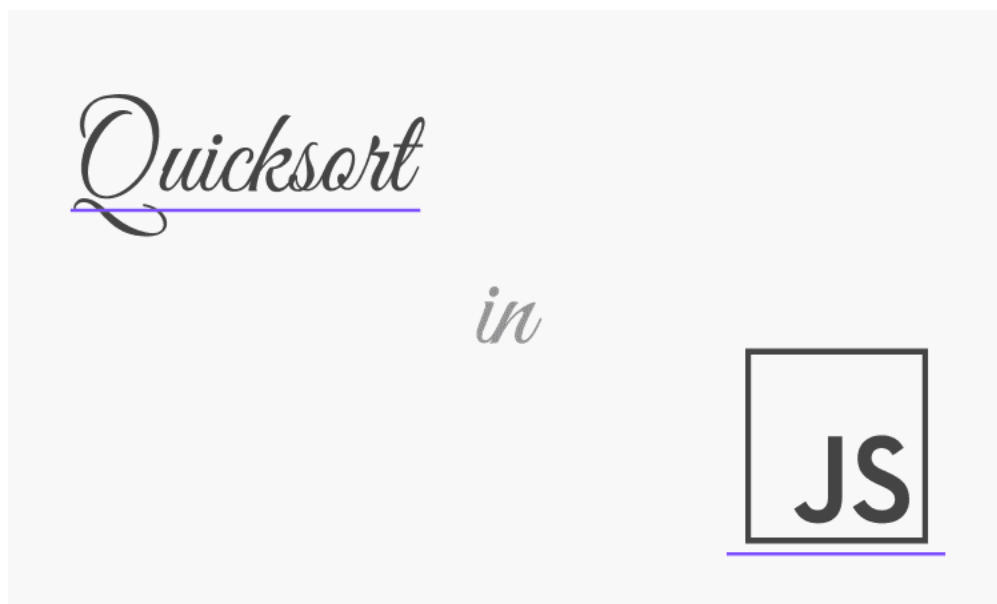
Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!



Search ...

Trang chủ > Sắp xếp > Thuật toán sắp xếp nhanh - C



Thuật toán sắp xếp nhanh – Quick Sort

📅 30 Jan, 2021 👤 Anh Nguyễn 💬 Leave a comment

NỘI DUNG BÀI VIẾT

Giới thiệu

Ý tưởng thuật toán sắp xếp nhanh

Bước 1

Bước 2

Bước 3

Bước 4

Bước 5

Sắp xếp nhanh sử dụng đệ quy để sắp xếp các mảng con

Chia

Trị

Kết hợp

Thuật toán sắp xếp nhanh

Cài đặt thuật toán sắp xếp nhanh

Đánh giá thuật toán sắp xếp nhanh

Ứng dụng của thuật toán sắp xếp nhanh

Giới thiệu

CODEGYM

**TỔNG HỢP
TÀI LIỆU LẬP TRÌNH HAY NHẤT
MIỄN PHÍ**

+150 bài thực hành

+ 50 tài liệu, sách

+ 30 video

+ 5 website

DANH MỤC

Angular

Bất đồng bộ

Biến, kiểu dữ liệu và toán tử

Bootstrap

Canvas

Cấu trúc dữ liệu và giải thuật

Cấu trúc điều kiện

Cấu trúc lặp

Clean Code & Refactoring

Cơ sở dữ liệu

CSS

Design Pattern

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!

Không phải tự nhiên mà nó được gọi là thuật toán sắp xếp nhanh. So với các thuật toán sắp xếp phổ biến khác thì có lẽ hiệu năng của Quick Sort sẽ vượt trội hơn rất nhiều so với Bubble Sort, Selection Sort, Insertion Sort,...

Ý tưởng thuật toán sắp xếp nhanh

Bước 1

Một phần tử chốt được chọn từ mảng. Bạn có thể chọn bất kỳ phần tử nào từ mảng làm phần tử chốt.

Ở đây, chúng ta chọn phần tử nằm ở vị trí ngoài cùng bên phải (tức là phần tử nằm ở cuối mảng) của mảng làm phần tử chốt.



Chọn phần tử chốt

Bước 2

[Tổng hợp 200+ tài liệu, sách, bài thực hành, video hướng dẫn lập trình... từ cơ bản đến nâng cao](#)

Các phần tử nhỏ hơn phần tử chốt được đặt ở bên trái và các phần tử lớn hơn phần tử chốt được đặt ở bên phải.



Đặt tất cả các phần tử nhỏ hơn ở bên trái và lớn hơn ở bên phải của phần tử chốt

Một con trỏ được cố định tại phần tử chốt. Phần tử chốt được so sánh với các phần tử bắt đầu từ chỉ mục đầu tiên. Nếu phần tử đó lớn hơn phần tử chốt thì đánh dấu nó, con trỏ thứ hai được đặt cho phần tử đó.

Bây giờ, phần tử chốt được so sánh với các phần tử khác (con trỏ thứ ba). Nếu phần tử đó nhỏ hơn phần tử chốt sẽ được hoán đổi với phần tử lớn hơn được tìm thấy trước đó.

Hàm

HTML

Hướng đối tượng

JSON

Kiểm thử

Mảng

Ngoại lệ

Node.js

Sắp xếp

String

Tổng quan

Học JavaScript
51 lượt thích

Thích Trang

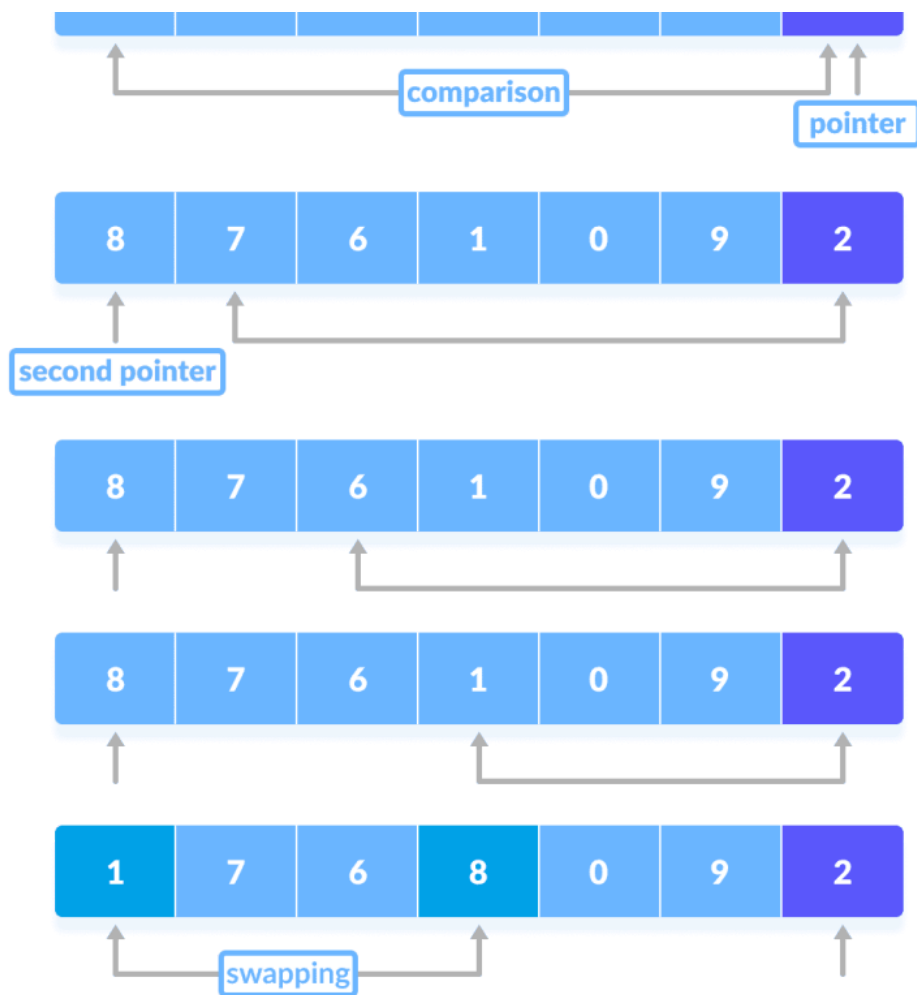
Học JavaScript
khoảng 2 tuần trước

Spread Operator có cú pháp giống với Rest Parameter tuy nhiên cả hai có ý nghĩa khác nhau. Rest Parameter được sử dụng khi khai báo hàm, ngược lại Spread Operator được sử dụng trong các câu lệnh, biểu thức hoặc khi gọi hàm.

ES6 cung cấp một toán tử mới gọi là Spread Operator bao gồm ba dấu chấm (...). Spread Operator cho phép bạn trải ra các phần tử của một đối tượng có thể lặp lại, chẳng hạn như một array, map hoặc set.

BÀI VIẾT NỔI BẬT

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript



So sánh phần tử chốt với các phần tử khác

Cuối cùng, phần tử chốt được hoán đổi với con trỏ thứ hai.

Tham khảo ngay!



Tỏ tình crush cùng ứng dụng Do You Love Me

Jingle Bell Jingle Bell, các bạn đã có crush để đi chơi Noel chưa nhỉ? Nếu chưa thì cũng đừng ... Continue reading



Học JavaScript

1



Ép kiểu trong JavaScript

Quá trình chuyển đổi từ kiểu dữ liệu này sang kiểu dữ liệu khác được gọi là chuyển đổi kiểu ... Continue reading



Học JavaScript

0



Top 10 ngôn ngữ lập trình đáng học nhất năm 2021 (phần 1)

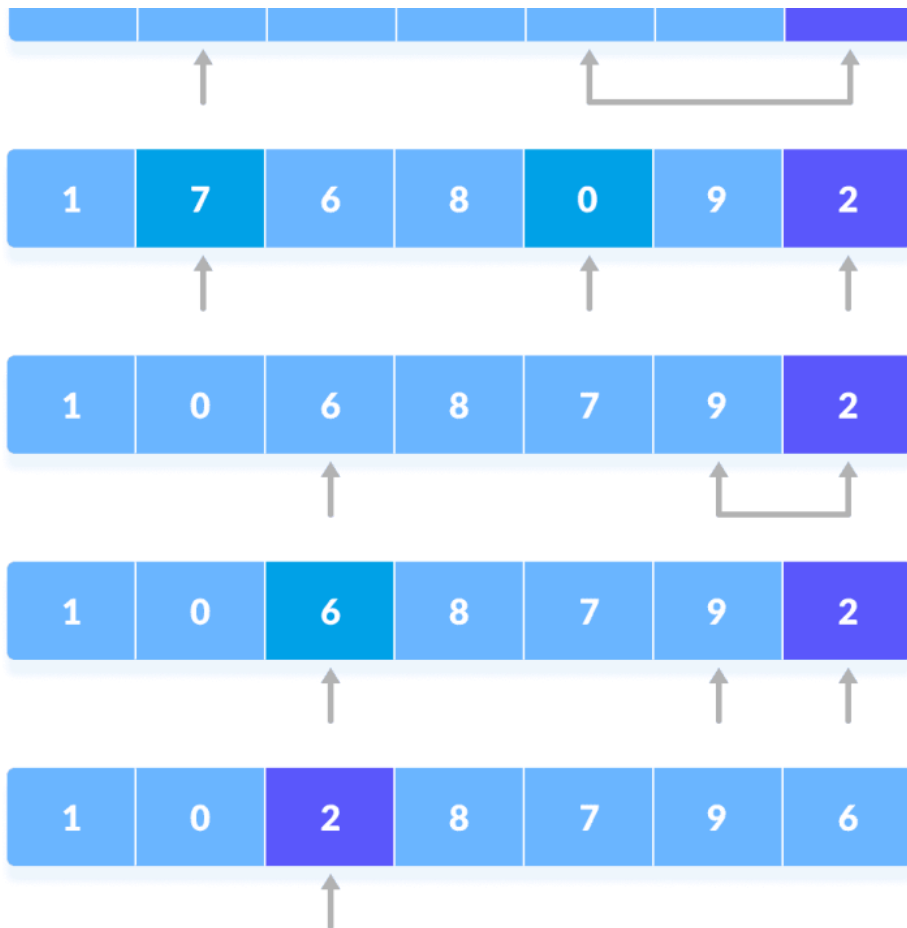
Hiện nay, có khoảng 600 ngôn ngữ lập trình trên mạng. Nhu cầu và sự phổ biến của các ngôn ... Continue reading



Học JavaScript

0

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript



Hoán đổi phần tử chốt với con trỏ thứ hai

Bây giờ các phần con bên trái và bên phải của phần tử chốt này được thực hiện để xử lý thêm trong các bước bên dưới.

Bước 3

Các phần tử chốt lại được chọn cho các phần tử con bên trái và bên phải. Trong các phần tử con này, phần tử chốt được đặt vào đúng vị trí của chúng. Sau đó, bước 2 được lặp lại.

Tham khảo ngay!



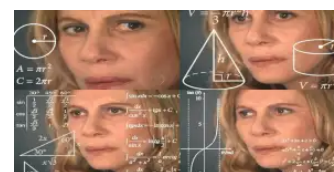
Top 10 laptop tốt nhất cho dân học lập trình

Do đặc thù công việc của ngành Công nghệ thông tin nên máy tính dành cho Lập trình viên cũng ... Continue reading



Học JavaScript

0



Dốt toán có thể học được lập trình không?

Xin chào các bạn, trong bài viết này mình sẽ chia sẻ quan điểm của mình về việc dốt toán ... Continue reading



Học JavaScript

2

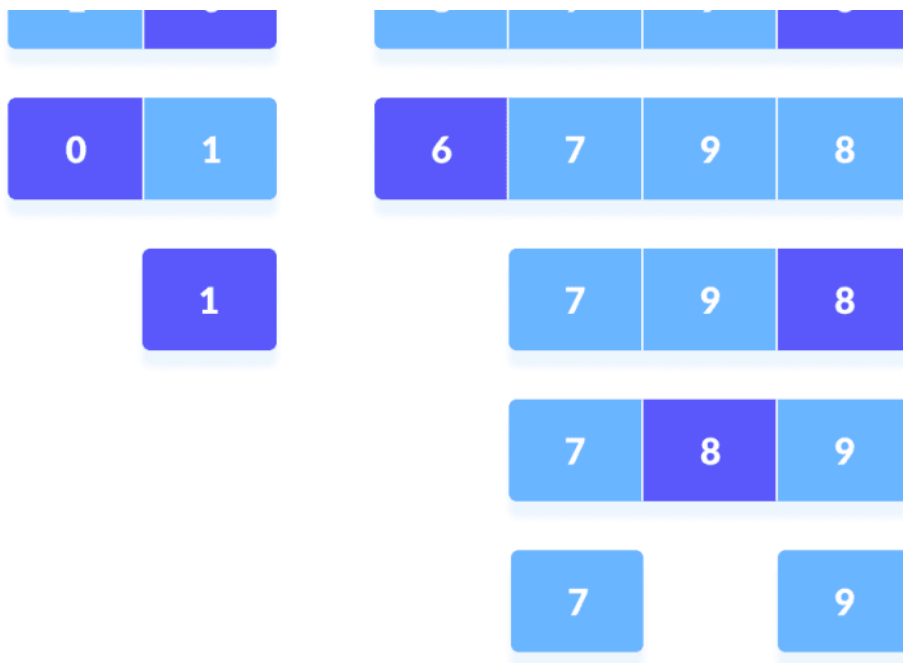
THAM KHẢO



TRANG WEB HỌC LẬP TRÌNH M
PHÍ

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!



Chọn phần tử chốt của mỗi nửa và đặt vào đúng vị trí bằng cách sử dụng đệ quy

Bước 4

Các phần con lại được chia thành các phần con nhỏ hơn cho đến khi mỗi phần con được tạo thành từ một phần tử duy nhất.

Bước 5

Tại thời điểm này, mảng đã được sắp xếp.

Sắp xếp nhanh sử dụng đệ quy để sắp xếp các mảng con

Dựa trên nguyên tắc của thuật toán **Thuật toán chia để trị**, thuật toán sắp xếp nhanh có thể được giải thích như sau:

Chia

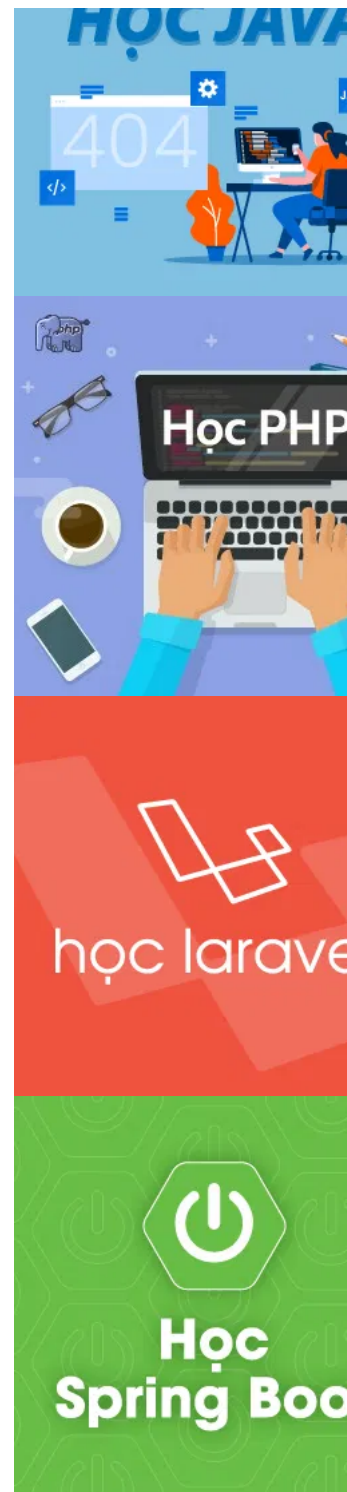
Mảng được chia thành các mảng con và lấy phần tử chốt làm điểm phân đoạn. Các phần tử nhỏ hơn phần tử chốt được đặt ở bên trái và các phần tử lớn hơn phần tử chốt được đặt ở bên phải.

Trị

Các phần con bên trái và bên phải lại được phân đoạn bằng cách chọn các phần tử chốt. Đệ quy cho đến khi không thể phân đoạn được nữa.

Kết hợp

Bước này không đóng vai trò quan trọng trong sắp xếp nhanh. Mảng đã được sắp xếp ở cuối bước trị.

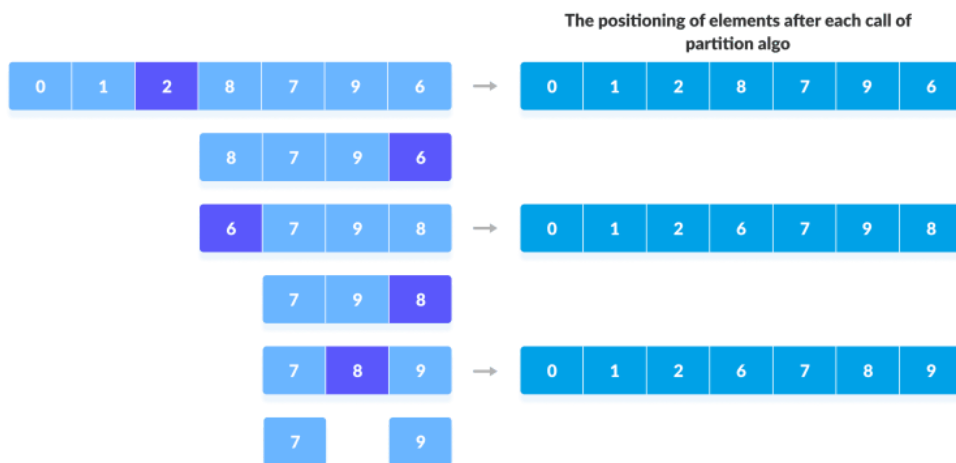


Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript



Sắp xếp các phần tử ở bên trái của phần tử chốt bằng cách sử dụng đệ quy

`quicksort(arr, pi+1, high)`



Sắp xếp các phần tử ở bên phải của phần tử chốt bằng cách sử dụng đệ quy

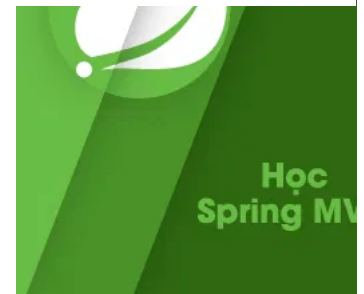
Thuật toán sắp xếp nhanh

```
quicksort(array, leftmostIndex, rightmostIndex)
if (leftmostIndex < rightmostIndex)
  pivotIndex <- partition(array, leftmostIndex, rightmostIndex)
  quicksort(array, leftmostIndex, pivotIndex)
  quicksort(array, pivotIndex + 1, rightmostIndex)
  partition(array, leftmostIndex, rightmostIndex)
  set rightmostIndex as pivotIndex
  storeIndex <- leftmostIndex - 1
  for i <- leftmostIndex + 1 to rightmostIndex
    if element[i] < pivotElement
      swap element[i] and element[storeIndex]
      storeIndex++
  swap pivotElement and element[storeIndex+1]
  return storeIndex + 1
```

Cài đặt thuật toán sắp xếp nhanh

```
function partition(array, low, high) {
  var pivot = array[high];
```

Tham khảo ngay!



TÀI LIỆU HỌC LẬP TRÌNH MIỄN

Cẩm nang Lập trình căn bản cho người mới bắt đầu



CodeGym Pages

Tổng hợp kiến thức Java căn bản

Tài liệu THINK JAVA cho người mới học



CodeGym Pages



Tài liệu Python cơ bản – Python cho người mới bắt đầu

CodeGym Pages

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!

```

i++;
var temp = array[i];
array[i] = array[j];
array[j] = temp;
}
}
temp = array[i + 1];
array[i + 1] = array[high];
array[high] = temp;
return i + 1;
}

function quickSort(array, low, high) {
  if (low < high) {
    var pi = partition(array, low, high);
    quickSort(array, low, pi - 1);
    quickSort(array, pi + 1, high);
  }
}

var arr = [3, 5, -2, 14, -9, 30];
quickSort(arr, 0, arr.length - 1);
console.log(arr);

```

Kết quả

```
[-9, -2, 3, 5, 14, 30]
```

Đánh giá thuật toán sắp xếp nhanh

Độ phức tạp thời gian:

Trường hợp xấu nhất: $O(n^2)$

Nó xảy ra khi phần tử chốt được chọn là phần tử lớn nhất hoặc nhỏ nhất. Điều kiện này dẫn đến trường hợp phần tử pivot nằm ở cuối cùng của mảng đã được sắp xếp. Một mảng con luôn trống và một mảng con khác chứa $n - 1$ phần tử. Do đó, Quick Sort chỉ được gọi trên mảng con này.

Trường hợp tốt nhất: $O(n \cdot \log n)$

Nó xảy ra khi phần tử chốt luôn là phần tử giữa hoặc gần phần tử giữa.

Trường hợp trung bình: $O(n \cdot \log n)$

Nó xảy ra khi các điều kiện trên không xảy ra.

Độ phức tạp không gian:

Độ phức tạp không gian của thuật toán sắp xếp nhanh là $O(\log n)$.

[Tham khảo: Khóa học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript](#)

Ứng dụng của thuật toán sắp xếp nhanh

Thuật toán sắp xếp nhanh được sử dụng trong các trường hợp:

Được sử dụng ở mọi nơi mà không cần đến sự ổn định

trang giấy



CodeGym Pages

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!

Các bạn có thể tham khảo các bài viết này về thuật toán sắp xếp trong JavaScript [tại đây](#).

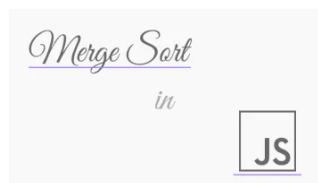
Hãy tham gia nhóm [Học lập trình](#) để thảo luận thêm về các vấn đề cùng quan tâm.

Chia sẻ

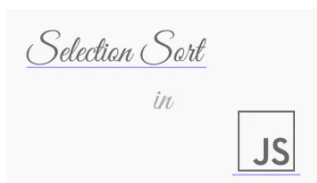
Share 0

Tweet

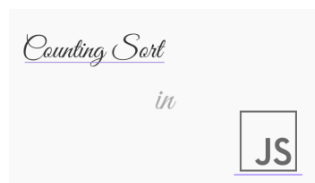
Bài viết liên quan



Thuật toán sắp xếp trộn - Merge Sort
In "Sắp xếp"



Thuật toán sắp xếp chọn - Selection Sort
In "Sắp xếp"



Thuật toán sắp xếp đếm phân phối - Counting Sort
In "Sắp xếp"



Xem thêm:

[Độ quy cho người mới bắt đầu](#)

[10 thủ thuật thú vị và hữu dụng trong JavaScript](#)

[Top 8 JS Framework không làm bạn thất vọng năm 2021](#)

Thuật toán sắp xếp trộn - Merge Sort

Thuật toán sắp xếp đếm phân phối - Counting Sort

Leave a Reply

Your email address will not be published. Required fields are marked *

Giảm **20%** khoá học Online nền tảng lập trình cho người mới bắt đầu bằng ngôn ngữ JavaScript

Tham khảo ngay!

Name*	Email*	Website
-------	--------	---------

☐ Save my name, email, and website in this browser for the next time I comment.

POST COMMENT



Copyright ©2021 Học JavaScript. Education Zone | Developed By Rara Theme. Powered by WordPress.