



# Bubble Sort

in



## Thuật toán sắp xếp nổi bọt – Bubble Sort

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

### NỘI DUNG BÀI VIẾT



Giới thiệu

Ý tưởng thuật toán sắp xếp nổi bọt

Bước 1

Bước 2

Thuật toán sắp xếp nổi bọt

Cài đặt thuật toán sắp xếp nổi bọt

Thuật toán sắp xếp nổi bọt tối ưu

Cài đặt thuật toán sắp xếp nổi bọt tối ưu

Đánh giá thuật toán sắp xếp nổi bọt

Ứng dụng của thuật toán sắp xếp nổi bọt

## Giới thiệu

Thuật toán sắp xếp nổi bọt (Bubble Sort) là thuật toán so sánh các phần tử liền kề và hoán đổi vị trí của chúng nếu chúng không theo thứ tự đã định. Thứ tự có thể tăng dần hoặc giảm dần.

## Ý tưởng thuật toán sắp xếp nổi bọt

### Bước 1

**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

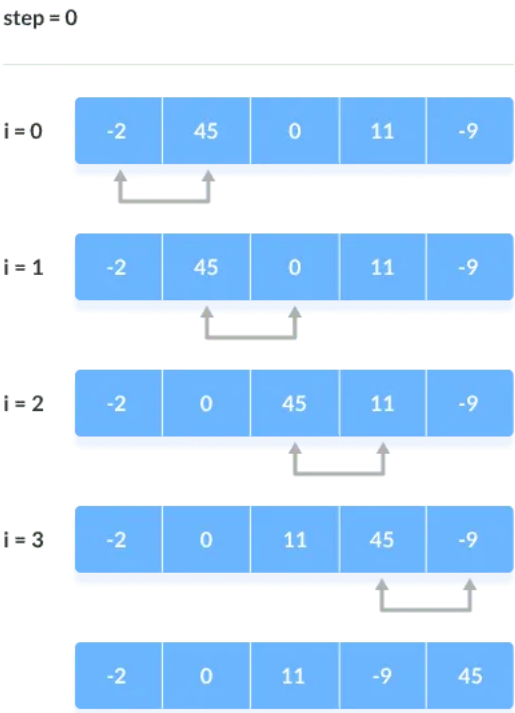
Design Pattern

DOM

Bắt đầu từ chỉ mục đầu tiên, so sánh phần tử đầu tiên và phần tử thứ hai. Nếu phần tử đầu tiên lớn hơn phần tử thứ hai, chúng sẽ được hoán đổi.

Bây giờ, hãy so sánh phần tử thứ hai và thứ ba. Trao đổi chúng nếu chúng không theo thứ tự.

Quá trình trên cứ tiếp tục cho đến phần tử cuối cùng.



Bước 2

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

Quá trình tương tự diễn ra cho các lần lặp còn lại. Sau mỗi lần lặp, phần tử lớn nhất trong số các phần tử chưa được sắp xếp được đặt ở cuối.

Trong mỗi lần lặp, việc so sánh diễn ra cho đến phần tử chưa được sắp xếp cuối cùng.

Mảng được sắp xếp khi tất cả các phần tử chưa được sắp xếp được đặt đúng vị trí của chúng.

ES6

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

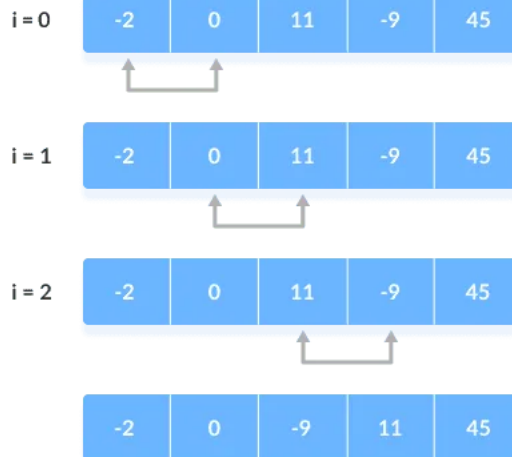
**Học JavaScript**  
khoảng 3 tuần trước

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.

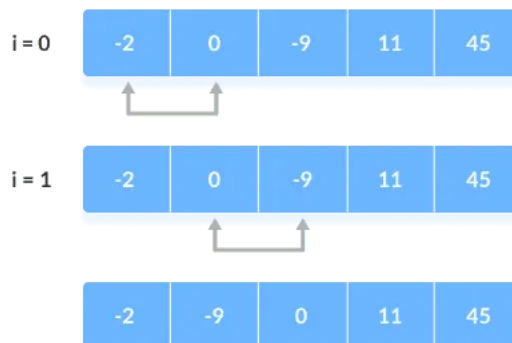
ES6 cung cấp một toán tử mới gọi là Spread Operator bao gồm ba dấu chấm (...). Spread Operator cho phép bạn trải ra các phần tử của một đối tượng có thể lặp lại, chẳng hạn như một array, map hoặc set.

BÀI VIẾT NỔI BẬT

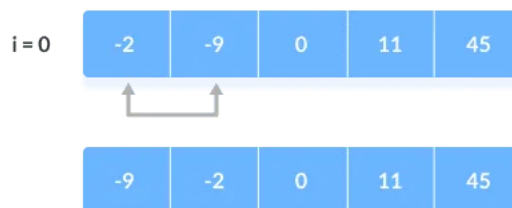
step = 1



step = 2



step = 3



## Thuật toán sắp xếp nổi bọt

```

bubbleSort(array)
for i <- 1 to indexOfLastUnsortedElement-1
  if leftElement > rightElement
    swap leftElement and rightElement
  end bubbleSort

```

## Cài đặt thuật toán sắp xếp nổi bọt



### 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

```
function bubbleSort(array) {
  var size = array.length;
  // run loops two times: one for walking through the array
  // and the other for comparison
  for (var i = 0; i < size - 1; i++) {
    for (var j = 0; j < size - i - 1; j++) {
      // To sort in descending order, change > to < in this line.
      if (array[j] > array[j + 1]) {
        // swap if greater is at the rear position
        var temp = array[j];
        array[j] = array[j + 1];
        array[j + 1] = temp;
      }
    }
  }
  var arr = [3, 5, -2, 14, -9, 30];
  bubbleSort(arr);
  console.log(arr);
}
```

Kết quả

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

## Thuật toán sắp xếp nổi bọt tối ưu

Trong đoạn mã trên, tất cả các so sánh có thể được thực hiện ngay cả khi mảng đã được sắp xếp. Nó làm tăng thời gian thực hiện.

Đoạn mã có thể được tối ưu hóa bằng cách thêm 1 biến **swapped**. Sau mỗi lần lặp, nếu không có sự hoán đổi nào diễn ra thì không cần thực hiện các vòng lặp khác.

Trong trường hợp này, biến **swapped** được đặt là false. Do đó, chúng ta có thể ngăn được lần lặp tiếp theo.

```
bubbleSort(array)
swapped <- false
for i <- 1 to indexOfLastUnsortedElement-1
  if leftElement > rightElement
    swap leftElement and rightElement
  swapped <- true
end bubbleSort
```

## Cài đặt thuật toán sắp xếp nổi bọt tối ưu

```
function bubbleSort(array) {
  var size = array.length;
  // run loops two times: one for walking through the array
  // and the other for comparison
  for (var i = 0; i < size - 1; i++) {
    // swapped keeps track of swapping
    var swapped = Boolean(true);
    for (var j = 0; j < size - i - 1; j++) {
      // To sort in descending order, change > to < in this line.
      if (array[j] > array[j + 1]) {
```



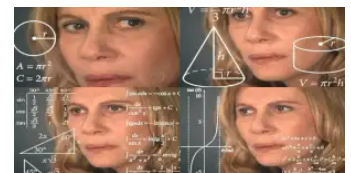
### Top 10 laptop tốt nhất cho dân học lập trình

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 MIỄN PHÍ

```
// swap if greater is at the rear position
var temp = array[j];
array[j] = array[j + 1];
array[j + 1] = temp;
swapped = false;
}
}
// If there is not swapping in the last swap, then the array is already sorted.
if (swapped === true)
break;
}
}
var arr = [3, 5, -2, 14, -9, 30];
bubbleSort(arr);
console.log(arr);
```

## Đánh giá thuật toán sắp xếp nổi bọt

Bubble Sort là một trong những thuật toán sắp xếp đơn giản nhất. Hai vòng lặp được thực hiện trong thuật toán.

Vòng lặp	Số lượng phép so sánh
1	(n-1)
2	(n-2)
3	(n-3)
.....	.....
Cuối cùng	1

Số lượng phép so sánh:  $(n - 1) + (n - 2) + (n - 3) + \dots + 1 = n(n - 1) / 2$  gần bằng  $n^2$ .

**Độ phức tạp:**  $O(n^2)$

Ngoài ra, chúng ta có thể phân tích độ phức tạp bằng cách quan sát số vòng lặp. Có 2 vòng lặp nên độ phức tạp là  $n * n = n^2$ .

**Độ phức tạp thời gian:**

**Trường hợp xấu nhất:**  $O(n^2)$

Nếu chúng ta muốn sắp xếp theo thứ tự tăng dần và mảng đã cho theo thứ tự giảm dần thì trường hợp xấu nhất sẽ xảy ra.

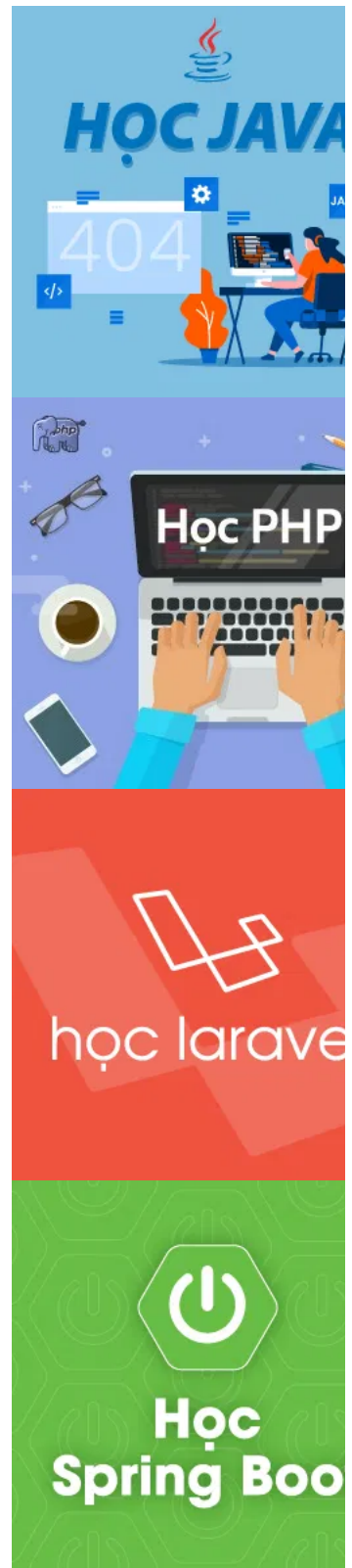
**Trường hợp tốt nhất:**  $O(n)$

Nếu mảng đã được sắp xếp thì không cần sắp xếp.

**Trường hợp trung bình:**  $O(n^2)$

Nó xảy ra khi các phần tử của mảng có thứ tự lộn xộn (không tăng dần cũng không giảm dần).

**Độ phức tạp không gian:**



Độ phức tạp không gian là  $O(1)$  vì một biến `temp` được sử dụng để hoán đổi.

Trong thuật toán sắp xếp nổi bọt tối ưu hóa, biến được hoán đổi làm tăng thêm độ phức tạp không gian, do đó, biến nó thành  $O(2)$ .

[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 nổi bọt

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

- Độ phức tạp của đoạn mã không quan trọng
- Ưu tiên đoạn mã ngắn

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

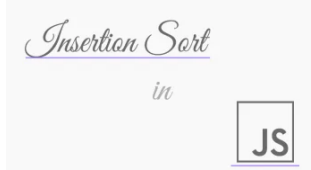

Hãy tham gia nhóm [Học lập trình](#) để thảo luận thêm về các vấn đề cùng quan tâm.

### Chia sẻ

Share 236

Tweet

#### Bài viết liên quan

 <p><i>Selection Sort</i> in JS</p>	 <p><i>Insertion Sort</i> in JS</p>	 <p><i>Quicksort</i> in JS</p>
Thuật toán sắp xếp chọn – Selection Sort In "Sắp xếp"	Thuật toán sắp xếp chèn – Insertion Sort In "Sắp xếp"	Thuật toán sắp xếp nhanh – Quick Sort In "Sắp xếp"



Xem thêm:

- [Đệ 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](#)

Sắp xếp trong JavaScript

Thuật toán sắp xếp chọn – Selection Sort



### 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

## Leave a Reply

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

Comment

Name\*

Email\*

Website

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

POST COMMENT

## Tổng hợp kiến thức PHP căn bản – Chỉ trong 01 trang giấy



 CodeGym Pages



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