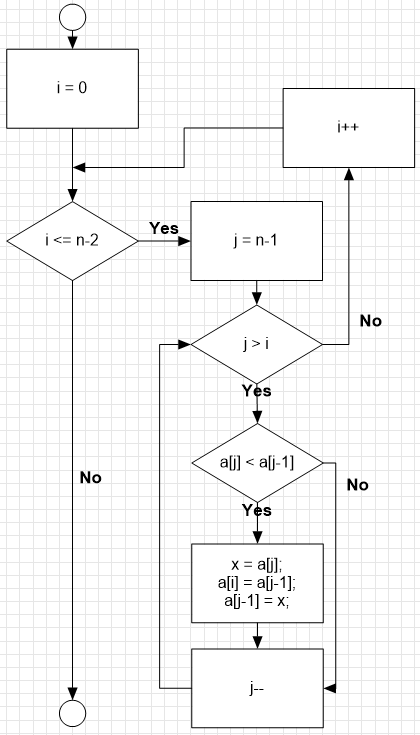
**Mục đích của sắp xếp**

* Sắp xếp làm cho các phần tử của mảng được sắp đặt theo một thứ tự nào đó nhằm thuận lợi cho quá trình thực hiện các thao tác khác.
* Ví dụ sắp xếp danh sách sinh viên theo điểm thi giảm dần
* Sắp xếp danh sách các tỉ phú trên thế giới theo tổng tài sản giảm dần
* Sắp xếp danh sách thí sinh theo tên abc

**Sắp xếp thủ công**

* Phần này ta sẽ tự viết thuật toán sắp xếp các phần tử của mảng. Thuật toán ta sử dụng ở đây là thuật toán sắp xếp nổi bọt.
* Các bước thực hiện của thuật toán gồm:
  + Cho i chạy từ 0 đến n – 2
  + Cho j chạy từ n – 1 đến i + 1
  + Nếu phần tử thứ j nhỏ hơn phần tử thứ j – 1:
    - Đổi chỗ hai phần tử này
  + Thuật toán sẽ đẩy dần các phần tử có giá trị nhỏ hơn về phía trước. Giống như hình ảnh bọt khí(phần tử nhỏ hơn) nhẹ hơn nước sẽ nổi lên mặt nước, phần tử nặng hơn sẽ chìm xuống(đưa về cuối mảng).
  + Sơ đồ khối của thuật toán:

[](https://braniumacademy.net/wp-content/uploads/2020/08/buble-sort.png)

* Thực thi thuật toán bằng chương trình:

public class BubbleSort {

    public static void main(String[] args) {

*// mảng gốc ban đầu*

int[] numbers = {9, 8, 5, 2, 3, 1, 0, 4, 7, 8, 9, 100, 7};

        System.*out*.println("Mảng gốc ban đầu: ");

*showArray*(numbers);

*// sắp xếp:*

*bubleSort*(numbers);

        System.*out*.println("Mảng sau khi sắp xếp: ");

*showArray*(numbers);

    }

*/\*\**

*\* phương thức sắp xếp các phần tử mảng theo thứ tự tăng dần*

*\* bằng thuật toán sắp xếp nổi bọt*

*\**

*\* @param numbers*

*\*/*

**private static void bubleSort(int[] numbers) {**

***// cho i chạy từ 0 đến n - 2***

**for (int i = 0; i < numbers.length - 1; i++) {**

***// cho j chạy từ n - 1 đến i + 1***

**for (int j = numbers.length - 1; j > i; j--) {**

**if (numbers[j] < numbers[j - 1]) { *// p.tử liền sau < liền trước***

**int x = numbers[j]; *// lưu lại phần tử liền sau***

**numbers[j] = numbers[j - 1]; *// cập nhật phần tử liền sau***

**numbers[j - 1] = x; *// cập nhật phần tử liền trước***

**}**

**}**

**}**

**}**

*/\*\**

*\* phương thức hiển thị các phần tử của mảng nguyên*

*\**

*\* @param numbers mảng chứa dữ liệu cần hiển thị*

*\*/*

private static void showArray(int[] numbers) {

        System.*out*.println("Các phần tử của mảng là: ");

        for (int i = 0; i < numbers.length; i++) {

            System.*out*.print(numbers[i] + " ");

        }

        System.*out*.println(); *// in xuống dòng*

}

}

Mảng gốc ban đầu:

Các phần tử của mảng là:

9 8 5 2 3 1 0 4 7 8 9 100 7

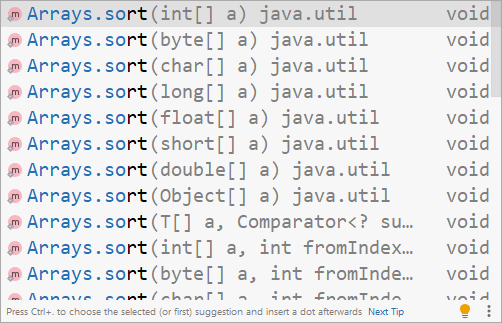
Mảng sau khi sắp xếp:

Các phần tử của mảng là:

0 1 2 3 4 5 7 7 8 8 9 9 100

**Sắp xếp quicksort**

* Lớp Arrays trong Java đã có sẵn một tập phương thức sắp xếp quicksort cho ta sử dụng:

[](https://braniumacademy.net/wp-content/uploads/2020/08/quick-sort.png)

* Bạn có thể sử dụng một trong các phương thức overload này để sắp xếp mảng. Ví dụ:

import java.util.Arrays;

public class BubbleSort {

    public static void main(String[] args) {

*// mảng gốc ban đầu*

int[] numbers = {9, 8, 5, 2, 3, 1, 0, 4, 7, 8, 9, 100, 7};

        System.*out*.println("Mảng gốc ban đầu: ");

*showArray*(numbers);

*// sắp xếp toàn bộ theo thứ tự tăng dần:*

**Arrays.*sort*(numbers);**

        System.*out*.println("Mảng sau khi sắp xếp: ");

*showArray*(numbers);

    }

*/\*\**

*\* phương thức hiển thị các phần tử của mảng nguyên*

*\**

*\* @param numbers mảng chứa dữ liệu cần hiển thị*

*\*/*

private static void showArray(int[] numbers) {

        System.*out*.println("Các phần tử của mảng là: ");

        for (int i = 0; i < numbers.length; i++) {

            System.*out*.print(numbers[i] + " ");

        }

        System.*out*.println(); *// in xuống dòng*

}

}

Mảng gốc ban đầu:

Các phần tử của mảng là:

9 8 5 2 3 1 0 4 7 8 9 100 7

Mảng sau khi sắp xếp:

Các phần tử của mảng là:

0 1 2 3 4 5 7 7 8 8 9 9 100

* Sắp xếp một đoạn trong mảng: tham số thứ hai là vị trí bắt đầu, tham số thứ ba là vị trí kết thúc sắp xếp.

import java.util.Arrays;

public class BubbleSort {

    public static void main(String[] args) {

*// mảng gốc ban đầu*

int[] numbers = {9, 8, 5, 2, 3, 1, 0, 4, 7, 8, 9, 100, 7};

        System.*out*.println("Mảng gốc ban đầu: ");

*showArray*(numbers);

*// sắp xếp toàn bộ theo thứ tự tăng dần:*

**Arrays.*sort*(numbers, 2, 8);**

        System.*out*.println("Mảng sau khi sắp xếp: ");

*showArray*(numbers);

    }

*/\*\**

*\* phương thức hiển thị các phần tử của mảng nguyên*

*\**

*\* @param numbers mảng chứa dữ liệu cần hiển thị*

*\*/*

private static void showArray(int[] numbers) {

        System.*out*.println("Các phần tử của mảng là: ");

        for (int i = 0; i < numbers.length; i++) {

            System.*out*.print(numbers[i] + " ");

        }

        System.*out*.println(); *// in xuống dòng*

}

}

Mảng gốc ban đầu:

Các phần tử của mảng là:

9 8 5 2 3 1 0 4 7 8 9 100 7

Mảng sau khi sắp xếp:

Các phần tử của mảng là:

9 8 0 1 2 3 4 5 7 8 9 100 7