Mục đích

Luyện tập cài đặt thuật toán sắp xếp nổi bọt.

Mô tả bài toán

Viết phương thức bubbleSort(int[] list) để sắp xếp các số trong một mảng theo trật tự tăng dần.

Hướng dẫn thực hiện

**Bước 1:** Tạo lớp BubbleSort, khai báo một mảng số nguyên chưa được sắp xếp.

**public class** BubbleSort {  
  
 **static int**[] list = {2, 3, 2, 5, 6, 1, -2, 3, 14, 12};

// codes below here  
  
}

**Bước 2:**Cài đặt phương thức BubbleSort (**int**[] list).

Sắp xếp nổi bọt là một giải thuật sắp xếp đơn giản. Giải thuật sắp xếp này được tiến hành dựa trên việc so sánh cặp phần tử liền kề nhau và tráo đổi thứ tự nếu chúng không theo thứ tự.

Giải thuật sắp xếp nổi bọt so sánh mỗi cặp phần tử trong mảng trừ khi cả toàn bộ mảng đó đã hoàn toàn được sắp xếp theo thứ tự tăng dần. Điều này có thể làm tăng độ phức tạp, tức là tăng các thao tác so sánh và tráo đổi không cần thiết nếu như mảng này không cần sự tráo đổi nào nữa khi tất cả các phần tử đã được sắp xếp theo thứ tự tăng dần rồi.

Để tránh việc này xảy ra, chúng ta có thể sử dụng một biến **needNextPass** chẳng hạn để giúp chúng ta biết có cần thực hiện thao tác tráo đổi thứ tự hay không. Nếu không cần thiết thì thoát khỏi vòng lặp.

**public static void** bubbleSort(**int**[] list) {  
 **boolean** needNextPass = **true**;  
  
 **for** (**int** k = 1; k < list.**length** && needNextPass; k++) {  
 *// Array may be sorted and next pass not needed* needNextPass = **false**;  
 **for** (**int** i = 0; i < list.**length** - k; i++) {  
 **if** (list[i] > list[i + 1]) {  
 *// Swap list[i] with list[i + 1]* **int** temp = list[i];  
 list[i] = list[i + 1];  
 list[i + 1] = temp;  
  
 needNextPass = **true**; *// Next pass still needed* }  
 }  
 }  
}

**Bước 3:**Cài đặt phương thức main để thực thi ứng dụng.

**public static void** main(String[] args) {  
 *bubbleSort*(list);  
 **for** (**int** i = 0; i < list.**length**; i++)  
 System.***out***.print(list[i] + **" "**);  
}

**Bước 4:** Chạy chương trình. Quan sát kết quả trả về.