

Homework 2: SimpleSort

- Bài 1.** Viết chương trình sắp xếp dãy số nguyên sử dụng thuật toán sắp xếp nổi bọt. In ra trạng thái của dãy số sau mỗi vòng lặp để thấy được sự thay đổi. Đếm số lần so sánh và số lần đổi chỗ.
- Bài 2.** Viết chương trình sắp xếp dãy số nguyên sử dụng thuật toán sắp xếp chọn. In ra trạng thái của dãy số sau mỗi vòng lặp để thấy được sự thay đổi. Đếm số lần so sánh và số lần đổi chỗ.
- Bài 3.** Viết chương trình sắp xếp dãy số nguyên sử dụng thuật toán sắp xếp chèn. In ra trạng thái của dãy số sau mỗi vòng lặp để thấy được sự thay đổi. Đếm số lần so sánh và số lần đổi chỗ.
- Bài 4.** Chạy các chương trình ở bài 1, 2, 3 với đầu vào là 1 dãy đã được sắp xếp. Thuật toán nào cho hiệu quả tốt hơn (số lần so sánh và đổi chỗ ít hơn).
- Bài 5.** Làm các bài tập 1, 2, 3 nhưng với dữ liệu tổng quát để có thể sắp xếp mọi dãy đối tượng T có giao diện Comparable<T>.
- Bài 6.(*)** Tạo đối tượng Card (quân bài) gồm 2 thuộc tính là rank và suit. Tạo bộ bài gồm 52 quân bài.
- Tạo đối tượng **comparecard** so sánh 2 quân bài với nhau Comparator<Card> (giao diện Comparator). Sử dụng thư viện Arrays, phương thức sort(T[] a, Comparator<? super T> c) để sắp xếp bộ bài. (a là bộ bài, c là đối tượng **comparecard**).
- Cài đặt giao diện Comparable cho đối tượng Card, sử dụng các phương thức sắp xếp đã làm ở bài 5 để sắp xếp bộ bài.
- Bài 7.(*)** Sử dụng đối tượng Random, tạo dãy N số nguyên ngẫu nhiên, mỗi số $\leq M$ và không có 2 số nào bằng nhau. ($N < M$).
- Tạo ra bộ 52 quân bài. Trộn bộ bài vừa tạo.