Câu 1: Trong các thuật toán sắp xếp đã học, phương pháp quicksort tối ưu nhất. Kém tối ưu nhất là Selection sort. Đánh giá dựa vào độ phức tạp của 2 phương pháp.

Câu 2: cả 2 phương pháp đều như nhau khi:

-Trường hợp nếu dãy đó đã được sắp xếp từ đầu và trường hợp dãy đó chỉ có 1 phần tử thì cả 2 phương pháp tìm kiếm đã học đều như nhau. Vì:

+Dãy đã được sắp xếp thì tìm tuần tự và nhị phân xem là 1, bản chất tìm kiếm nhị phân là sắp xếp rồi tìm và tuần tự thì tìm từ đầu.

+Dãy có 1 phần tử thì cả 2 phương pháp tìm như nhau.

Câu 3: Sắp xếp cơ số (RadixSort)

* Đầu tiên, thuật toán sẽ chia các phần tử thành các nhóm, dựa trên chữ số cuối cùng (hoặc dựa theo bit cuối cùng, hoặc vài bit cuối cùng).
* Sau đó ta đưa các nhóm lại với nhau, và được danh sách sắp xếp theo chữ số cuối của các phần tử. Quá trình này lặp đi lặp lại với chữ số át cuối cho tới khi tất cả vị trí chữ số đã sắp xếp.

Ưu điểm: Có thể chạy nhanh hơn các thuật toán sắp xếp sử dụng so sánh. Ví dụ nếu ta sắp xếp các số nguyên 32 bit, và chia nhóm theo 1 bit, thì độ phức tạp là O(N). Trong trường hợp tổng quát, độ phức tạp là O(N∗log(max(ai))).

Nhược điểm: Không thể sắp xếp số thực.