**NGUYỄN THẾ HOÀNG – IT81**

**MSSV : 1851050048**

**NHÓM 3**

***Câu 1: Trong các phương pháp xếp thứ tự đã học, phương pháp nào tối ưu nhất, và kém tối ưu nhất ? Tại sao ?***

* Phương pháp tối ưu nhất là Quick Sort.

+ Vì độ phức tạp của thuật toán Quick Sort là O(n logn) và trường hợp xấu nhất là ***O(n2 ).***

* Phương pháp kém tối ưu nhất là Bubble Sort.

+ Vì độ phức tạp của thuật toán Bubble Sort là O(n2) và trường hợp xấu nhất là

***n(n – 1)/ 2.***

***Câu 2: Trong 2 phương pháp tìm kiếm đã học, trường hợp nào thì cả 2 phương pháp đều như nhau ? Giải thích tại sao ?***

* Trong các phương pháp đã học, 2 phương pháp đều như nhau là ***Selection Sort và Bubble Sort.***
* Vì cả hai trường hợp độ phưc tạp đều giống nhau và ở trưởng hợp tốt nhất độ phức tạp của 2 phương pháp đều là ***n(n – 1)/2.***

***Câu 3: Ngoài các phương pháp xếp thứ tự đã học, hãy tìm hiểu them 1 phương pháp xếp thứ tự khác, giới thiệu và giải thích***

* Ngoài các phương pháp đã học thì còn có một số phương pháp khác như : Shell Sort, Radix Sort, Merge Sort,…

Giới thiệu về Merge Sort:

Giống như Quick sort, Merge sort là một thuật toán chia để trị. Thuật toán này chia mảng cần sắp xếp thành 2 nửa. Tiếp tục lặp lại việc này ở các nửa mảng đã chia. Sau cùng gộp các nửa đó thành mảng đã sắp xếp. Hàm merge() được sử dụng để gộp hai nửa mảng. Hàm merge(arr, l, m, r) là tiến trình quan trọng nhất sẽ gộp hai nửa mảng thành 1 mảng sắp xếp, các nửa mảng là arr[l…m] và arr[m+1…r] sau khi gộp sẽ thành một mảng duy nhất đã sắp xếp.

Độ phức tạp của thuật toán:

+ Trường hợp tốt nhất: ***O(n log n).***

+ Trường hợp trung bình: ***O(n log n).***

+ Trường hợp xấu nhất: ***O(n log n).***