

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH



BÁO CÁO KẾT QUẢ THỬ NGHIỆM
MÔN: CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT (IT003)
ĐỀ TÀI: THỰC NGHIỆM CÁC GIẢI THUẬT SẮP XẾP NỘI

Giảng viên hướng dẫn: **Nguyễn Thanh Sơn**

Sinh viên thực hiện: **Lại Đức Huy - 25520691**

Lớp: **IT003.Q21.TTNT**

Khóa: **K20**

TP. Hồ Chí Minh, tháng 2 năm 2026

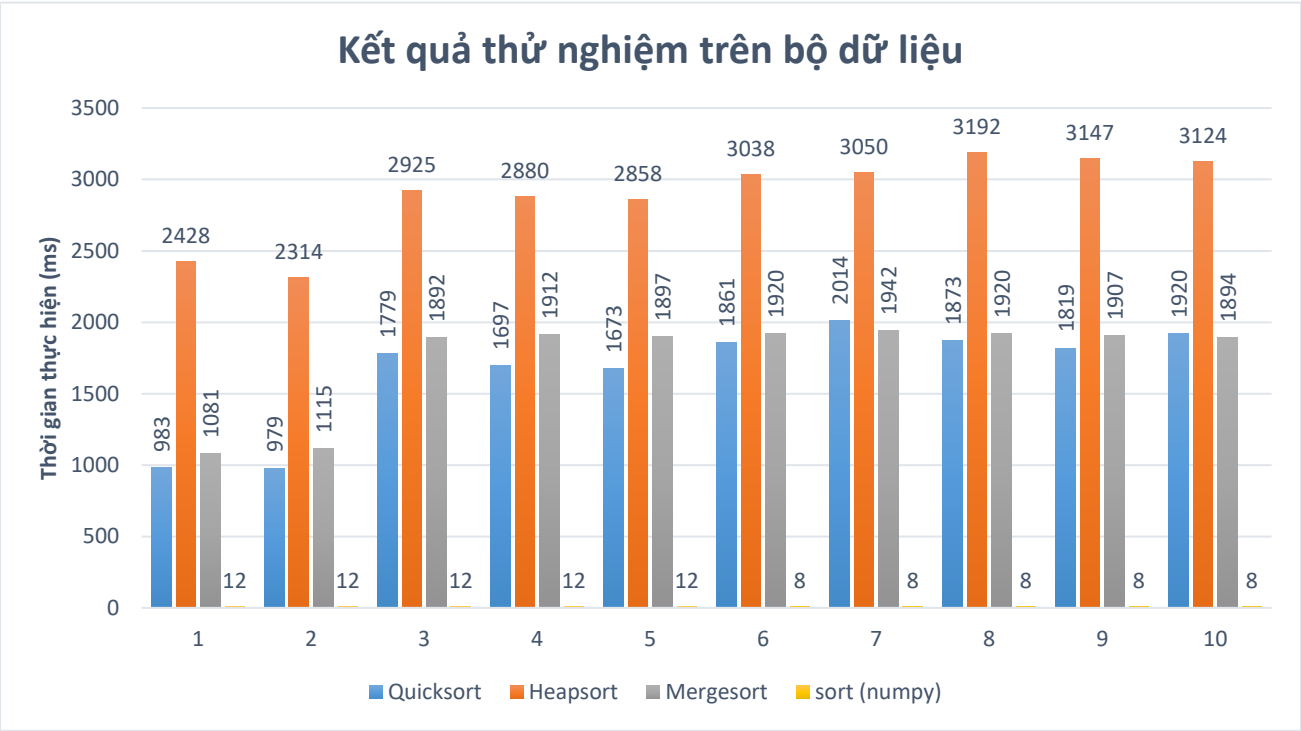
BÁO CÁO KẾT QUẢ THỬ NGHIỆM

I. Kết quả thử nghiệm

1. Bảng thời gian thực hiện¹

Dữ liệu	Thời gian thực hiện (ms)			
	Quicksort	Heapsort	Mergesort	sort (numpy)
1	983	2428	1081	12
2	979	2314	1115	12
3	1779	2925	1892	12
4	1697	2880	1912	12
5	1673	2858	1897	12
6	1861	3038	1920	8
7	2014	3050	1942	8
8	1873	3192	1920	8
9	1819	3147	1907	8
10	1920	3124	1894	8
Trung bình	1659.8	2895.6	1748	10

2. Biểu đồ (cột) thời gian thực hiện



II. Kết luận:

1. Tổng thể:

- Kết quả cho thấy thuật sắp xếp nhanh nhất là numpy sort, trung bình khoảng 10ms, nhanh hơn 160-290 lần so với những thuật toán tự cài đặt bằng Python.
- Trong các thuật toán còn lại, Quicksort có tốc độ nhanh nhất (trung bình 1659.8ms), Mergesort (trung bình 1748ms) và chậm nhất là Heapsort (2895.6ms).

¹ Số liệu chỉ mang tính minh họa

2. *Đánh giá các thuật toán tự cài đặt:*

- **Quicksort:** tuy thời gian hoàn thành là nhanh nhất trong các thuật toán tự cài đặt (trung bình khoảng 1659.8ms), nhưng Quicksort kém ổn định nhất (thời gian dao động từ 979ms đến 2014ms, chênh nhau 1035ms). Điều này là vì tùy vào dữ liệu đầu vào và cách chọn pivot (điểm chốt) mà có thể dẫn đến trường hợp xấu nhất với độ phức tạp $O(n^2)$ khiến thời gian hoàn thành chậm hơn.
- **Mergesort:** là thuật toán nhanh thứ 2 với thời gian hoàn thành trung bình là 1748ms, Mergesort là thuật toán hoạt động ổn định nhất (thời gian dao động từ 1080ms đến 1942ms, chênh nhau 861ms). Điều này đúng với độ phức tạp lý thuyết không thay đổi là $O(n \log n)$.
- **Heapsort:** là thuật toán tự cài đặt chậm nhất (trung bình 2895.6ms). Điều này được giải thích vì Heapsort thực hiện hàm Heapify nhiều lần hoán đổi phần tử xa nhau trong mảng dẫn đến không tận dụng được cache máy tính nên chạy chậm hơn.

3. *Tóm tắt:*

- Numpy sort là thuật toán nhanh nhất do được triển khai bằng ngôn ngữ C trong thư viện, tối ưu bộ nhớ theo phần cứng máy tính. Do đó, trong thực tế khi làm việc trong môi trường Python hay những ngôn ngữ khác nên ưu tiên sử dụng những hàm có sẵn trong thư viện để đạt hiệu năng tốt nhất.
- Trong các thuật toán tự cài đặt, Quicksort cho thời gian hoàn thành trung bình nhanh nhất nhưng lại kém ổn định nhất vì hiệu năng phụ thuộc vào dữ liệu và cách chọn pivot. Mergesort cho thời gian ổn định và an toàn.

III. *Thông tin chi tiết:*

Link Github: <https://github.com/HuyDucUIT/Cac-giai-thuat-sap-xep-noi>