

BÁO CÁO KẾT QUẢ THỬ NGHIỆM

Sinh viên thực hiện: Đinh Nguyễn Bảo Anh

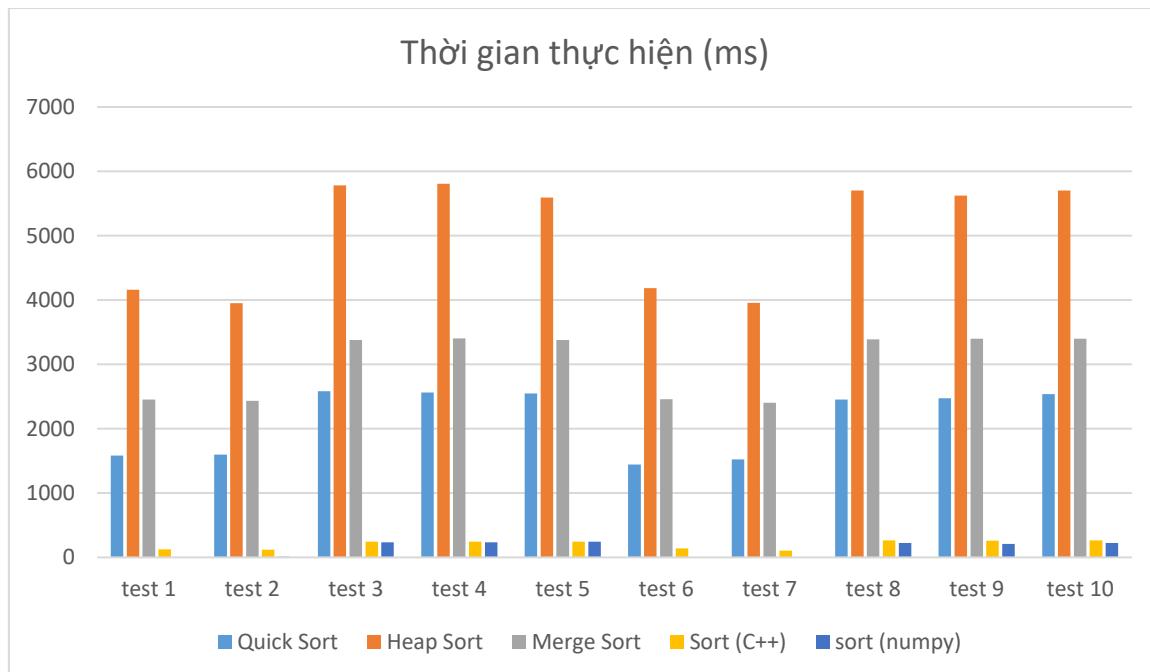
Nội dung báo cáo: So sánh các thuật toán sắp xếp cơ bản

I. Kết quả thử nghiệm

1. Bảng thời gian thực hiện

Dữ liệu	Thời gian thực hiện (ms)				
	Quicksort	Heapsort	Mergesort	sort (C++)	sort (numpy)
1	1582.708	4158.849	2453.232	125.000	7.186
2	1597.833	3952.760	2432.364	122.000	10.373
3	2581.242	5781.815	3381.179	244.00	236.735
4	2562.682	5805.833	3402.548	243.000	234.564
5	2547.212	5594.612	3380.730	245.000	243.700
6	1442.688	4182.665	2456.483	141.000	6.243
7	1525.185	3953.600	2404.913	105.000	7.750
8	2452.524	5703.038	3386.703	264.000	223.433
9	2474.827	5622.487	3398.702	261.000	210.002
10	2536.808	5701.079	3397.034	263.000	225.930
Trung bình	2130.371	5045.674	3009.389	201.300	140.592

2. Biểu đồ (cột) thời gian thực hiện



II. Kết luận:

- Với 10 bộ test gồm 5 bộ test số nguyên và 5 bộ test số thực (trong đó test 1 và 6 đã được sắp không giảm, test 2 và 7 đã được sắp không tăng, các test còn lại ngẫu nhiên) ta có nhận xét:

+ Các test 1, 2, 6, 7 nhìn chung cho kết quả khá tốt, chứng tỏ các thuật toán đều chạy nhanh đối với các dãy gần như được sắp xếp hoặc sắp xếp ngược.

+ Đặc biệt hai hàm Sort (C++) và Sort (numpy) chạy trong thời gian rất ngắn (Sort C++ ~ 200ms, Sort numpy ~ 140ms)

+ Các thuật toán còn lại chạy trong thời gian $O(n \log n)$. Trong đó Quick Sort với pivot random cho thời gian chạy ngắn nhất, tiếp đến là Merge Sort và cuối cùng là Heap Sort. Vì pivot random nên Quick Sort dễ dàng tránh được các trường hợp xấu, dẫn tới thời gian chạy nhanh hơn.

+ Cần lưu ý rằng với Quick Sort pivot cố định (thường là phần từ đầu hoặc cuối dãy) thì chạy với độ phức tạp thời gian $O(n^2)$ với các test 1, 2, 6, 7 khiến khối lượng tính toán có thể đạt $\sim 10^{12}$ bước. Do đó cần đánh giá dữ liệu để chọn pivot phù hợp.

- Về tính hiệu quả thì các thuật toán Sort (C++) hay Sort (numpy) là tối ưu nhất về mặt thời gian.

III. Thông tin chi tiết – link github, trong repo gibub cần có

https://github.com/bazxz137zxz-lab/25520042_Sorting_Report.git

Repo gồm:

- File báo cáo (25520042_Sorting_Report)

- Mã nguồn sinh test (src_sinh_test/gentest.py)

- Mã nguồn chạy test (src_sort/testcode.py và src/testcode.cpp)

(Dữ liệu thử nghiệm được sinh từ file src_sinh_test/gentest.py)