

Báo cáo kết quả thực nghiệm

Thực nghiệm các giải thuật sắp xếp nội

1. Giới thiệu

Báo cáo này trình bày kết quả **thực nghiệm các giải thuật sắp xếp nội** nhằm đánh giá thời gian thực thi của thuật toán trên các tập dữ liệu lớn có đặc điểm khác nhau. Mục tiêu của thực nghiệm là kiểm chứng lại **độ phức tạp lý thuyết** thông qua kết quả thực tế.

2. Môi trường thực nghiệm

- Ngôn ngữ lập trình: **Python**
- Phương pháp đo thời gian: `time.perf_counter()`
- Hệ điều hành: Windows
- Thiết bị: Máy tính cá nhân

3. Dữ liệu thực nghiệm

3.1 Mô tả tập dữ liệu

- Tổng số tập dữ liệu: **10**
- Số phần tử mỗi tập: **khoảng 1.000.000**
- Kiểu dữ liệu:
 - 5 tập số thực (1 đến 5)**
 - 5 tập số nguyên (6 đến 10)**

3.2 Đặc điểm dữ liệu

Tập dữ liệu	Mô tả
Data 1	Dữ liệu đã sắp xếp tăng dần
Data 2	Dữ liệu sắp xếp giảm dần
Data 3–10	Dữ liệu ngẫu nhiên

Việc sử dụng nhiều dạng dữ liệu giúp đánh giá thuật toán trong các trường hợp **tốt nhất, xấu nhất và trung bình**.

4. Phương pháp thực nghiệm

Đối với mỗi tập dữ liệu:

- Đọc dữ liệu từ file (không tính thời gian I/O).
- Sao chép mảng để đảm bảo tính công bằng giữa các lần chạy.
- Gọi hàm sắp xếp.
- Đo thời gian thực thi bằng `time.perf_counter()`.

5. Kết quả thực nghiệm

5.1 Bảng kết quả thời gian (ms)

Dữ liệu	Quicksort	Heapsort	Mergesort	sort (NumPy)
1	1572	2910	1445	21
2	1490	2830	1358	10
3	2095	4861	2101	10
4	1955	4489	2180	11
5	2447	4519	3291	10
6	2620	4106	3105	14
7	4326	3784	3417	13
8	3956	4543	2219	14
9	3123	6019	3325	15
10	3341	5155	2833	13
Trung bình	2693	4322	2527	13

6. Nhận xét và phân tích

- Dữ liệu đã được sắp xếp tăng/giảm dần cho thời gian thực thi tốt hơn.
- Dữ liệu ngẫu nhiên cho kết quả ổn định, phản ánh đúng độ phức tạp trung bình của thuật toán.
- Thuật toán sắp xếp mặc định của NumPy được cài đặt ở mức C, do đó có hiệu năng cao và không gặp vấn đề tràn stack như cài đặt Quick Sort thuần Python.
- Thuật toán Quick Sort sử dụng **pivot ngẫu nhiên** để tránh trường hợp xấu nhất ở tập giảm/tăng dần tránh vấn đề tràn stack và độ phức tạp $O(N^2)$

Kết quả thực nghiệm phù hợp với phân tích độ phức tạp lý thuyết của các giải thuật sắp xếp nội.

7. Kết luận

Qua quá trình thực nghiệm có thể rút ra các kết luận:

- Đặc điểm dữ liệu đầu vào ảnh hưởng lớn đến hiệu năng của thuật toán sắp xếp.
- Việc sử dụng các thư viện tối ưu như NumPy mang lại hiệu suất vượt trội khi xử lý dữ liệu lớn.
- Thực nghiệm là bước quan trọng để kiểm chứng các kết quả lý thuyết đã học.