

Biểu đồ và kết luận.

Mục Lục

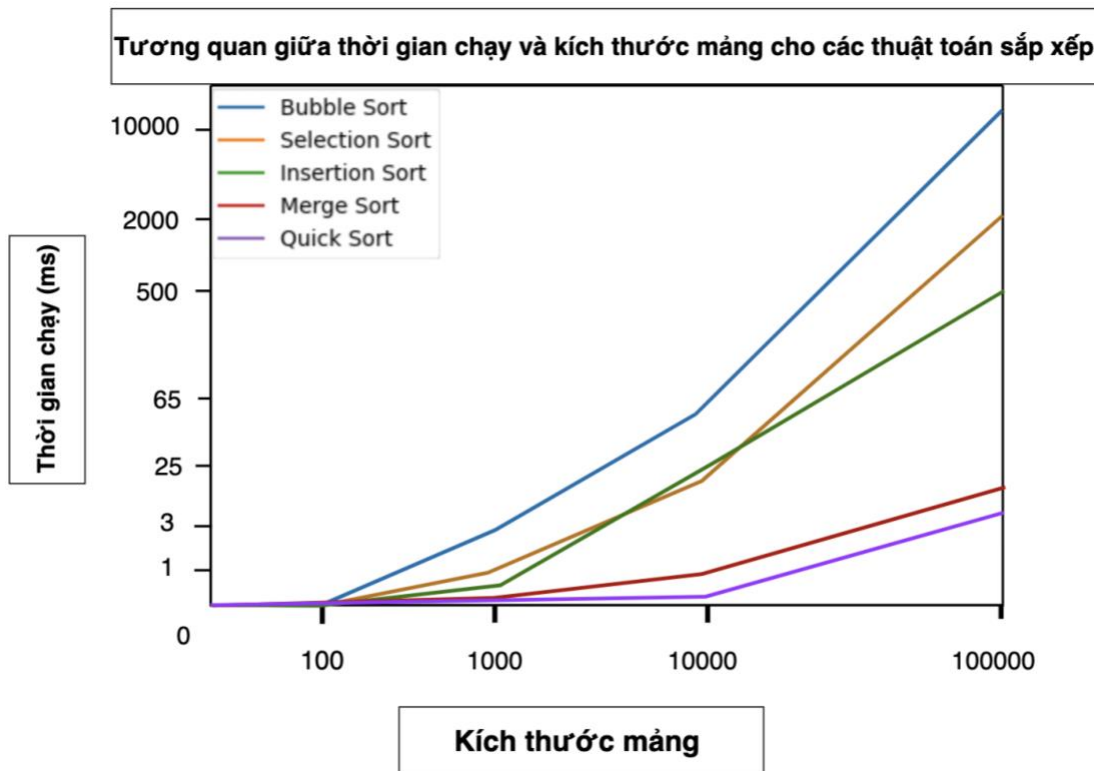
A, Biểu đồ.....	2
1, Biểu đồ so sánh các thuật toán sắp xếp dựa vào code bài tập 1 với ngôn ngữ java.	2
2, Biểu đồ so sánh các thuật toán sắp xếp code bằng ngôn ngữ python.	2
B, Kết luận.....	5
Sắp xếp nổi bọt (Bubble Sort)	5
Sắp xếp chọn (Selection Sort)	5
Sắp xếp chèn (Insertion Sort).....	5
Sắp xếp trộn (Merge Sort)	5
Sắp xếp nhanh (Quick Sort)	5
Tổng kết	5

A, Biểu đồ

1, Biểu đồ so sánh các thuật toán sắp xếp dựa vào code bài tập 1 với ngôn ngữ java.

Với phần code trong bài tập 1 với các kích thước mảng là random em đã dựa vào output để vẽ biểu đồ thể hiện tốc độ thời gian chạy và số lượng phần tử mảng tương ứng.

Dưới đây là biểu đồ của em :



Hình 1: Biểu đồ tốc độ thời gian chạy và số lượng phần tử mảng tương ứng.

2, Biểu đồ so sánh các thuật toán sắp xếp code bằng ngôn ngữ python.

Ngoài ra em cũng thử dùng ngôn ngữ python để mô tả tương quan giữa các thuật toán .

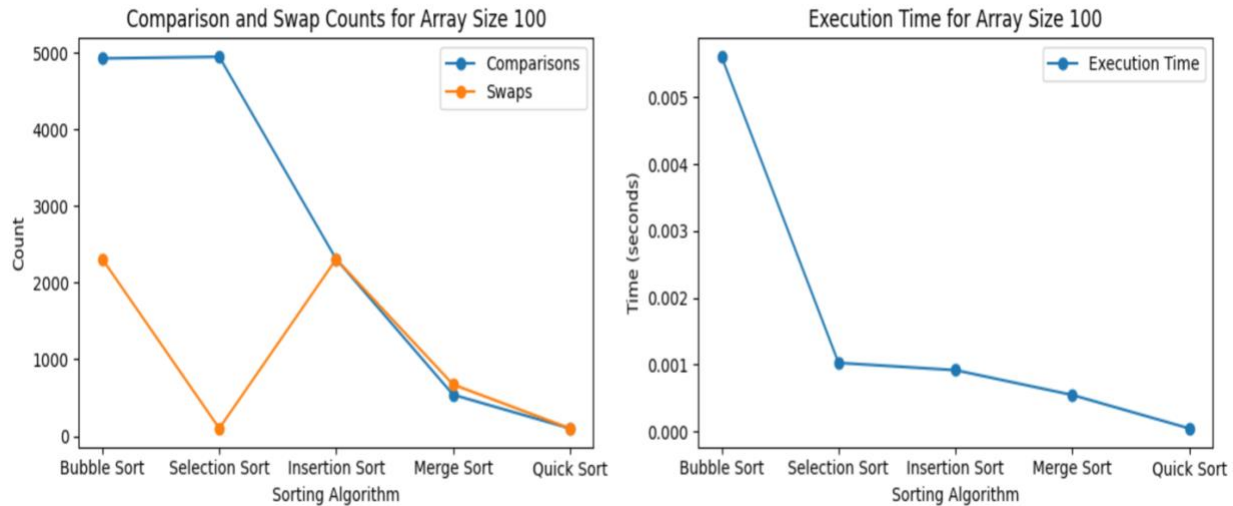
Link code bài làm của em ạ! :

❖ Bài code của hình 2,hình 3,hình 4, hình 5 :

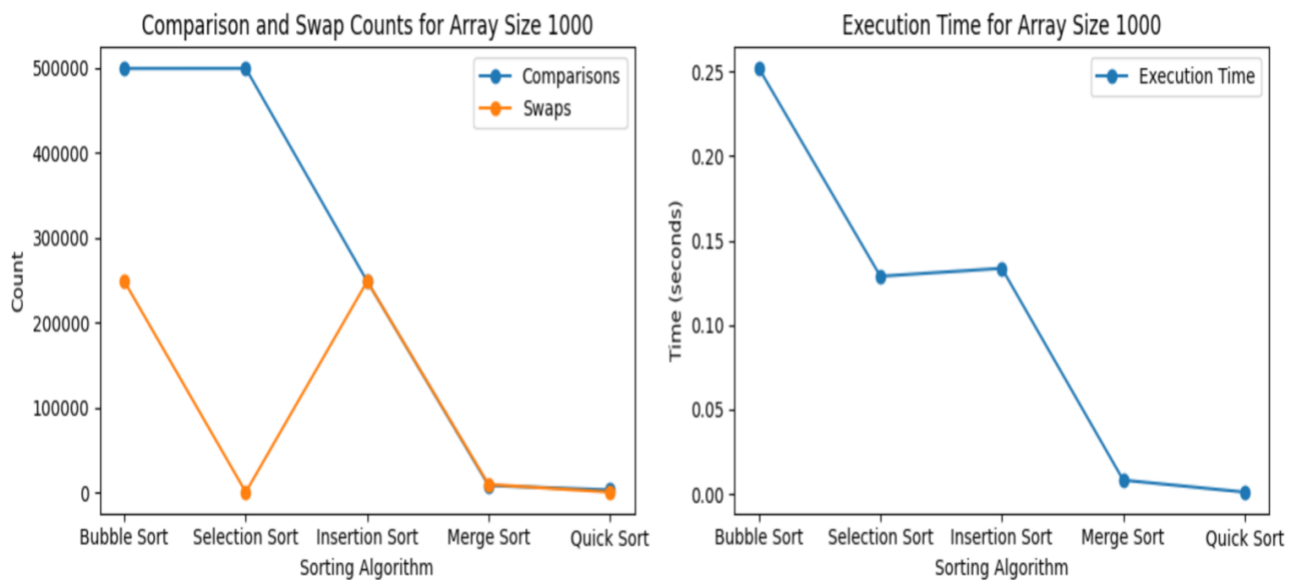
https://colab.research.google.com/drive/1-zOIHNUqongDPV9uxI7S_j-NEvmKfxSx?usp=sharing

Hình 2, hình 3, hình 4, hình 5 với :

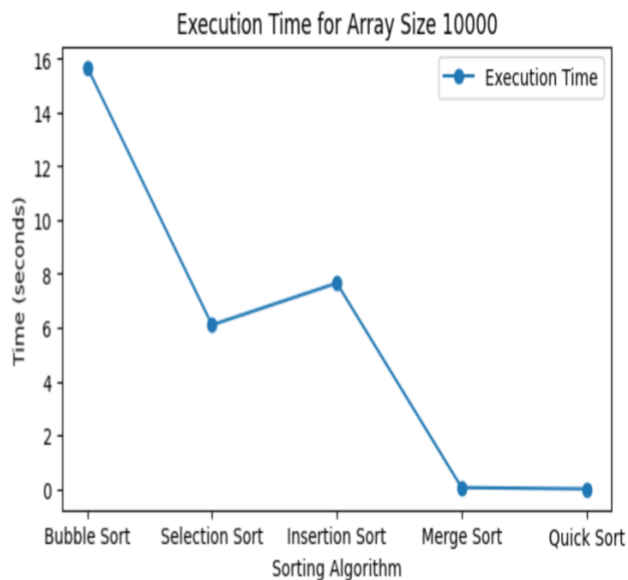
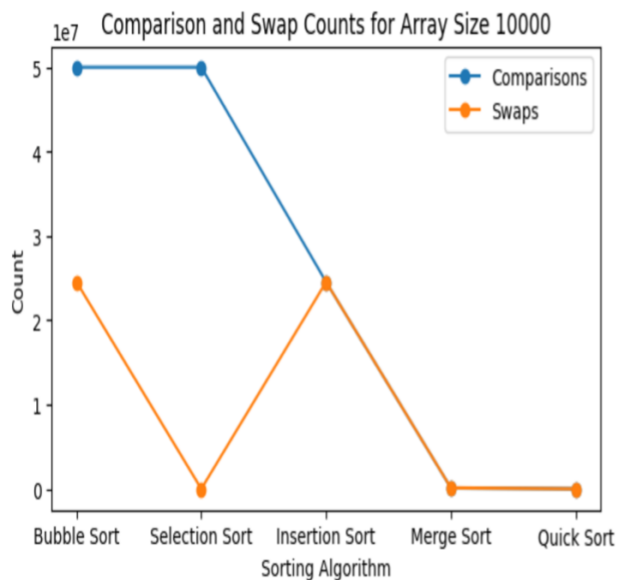
- ❖ Các hình bên trái là biểu thị tương quan số lần sắp xếp, số lần so sánh của mỗi thuật toán trong kích cỡ mảng N phần tử tương ứng.
- ❖ Các hình bên phải biểu thị tương quan thời gian chạy của mỗi thuật toán ứng với kích cỡ mảng N phần tử tương ứng.



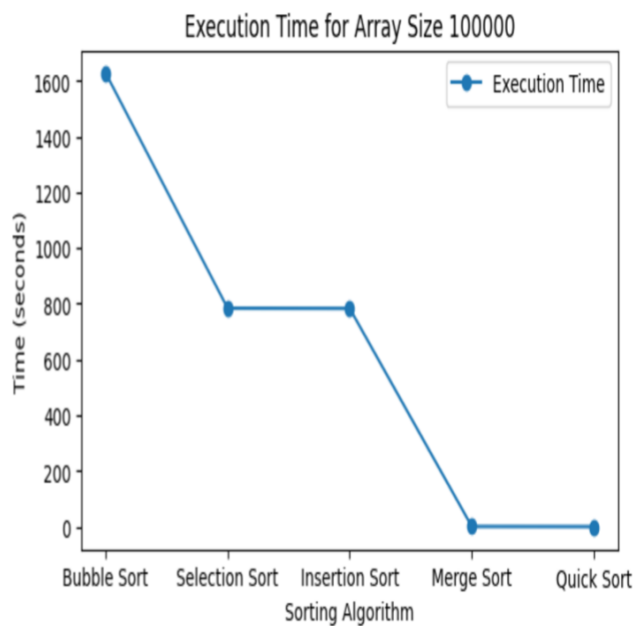
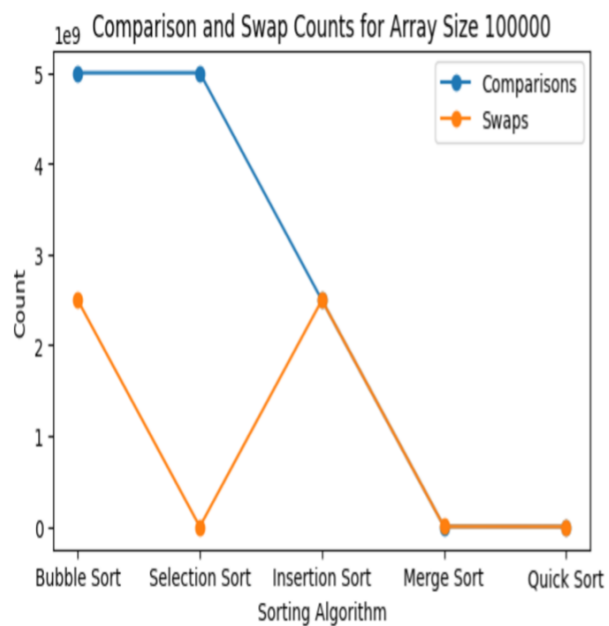
Hình 2: Kích thước mảng $N=100$ phần tử



Hình 3: Kích thước mảng $N=1000$ phần tử.



Hình 4: Kích thước mảng N=10000 phần tử.



Hình 5: Kích thước mảng N=100000 phần tử.

B, Kết luận

Dựa trên việc đánh giá hiệu suất các thuật toán sắp xếp với các giá trị N khác nhau và giá trị của miền dữ liệu là từ 1 đến 1.000.000, chúng ta có thể rút ra một số nhận xét và kết luận như sau:

Sắp xếp nổi bọt (Bubble Sort):

- Hiệu suất của thuật toán này không tốt, đặc biệt là với các dãy số lớn (N lớn).
- Sắp xếp nổi bọt thường không nên được sử dụng cho dãy số lớn hoặc dãy đã sắp xếp gần như hoàn chỉnh.

Sắp xếp chọn (Selection Sort):

- Hiệu suất của thuật toán này cũng không tốt, đặc biệt là với các dãy số lớn.
- Sắp xếp chọn không thể hiện sự cải thiện đáng kể so với sắp xếp nổi bọt.

Sắp xếp chèn (Insertion Sort):

- Sắp xếp chèn có hiệu suất tốt hơn so với sắp xếp nổi bọt và sắp xếp chọn, đặc biệt với các dãy số gần như đã sắp xếp.
- Tuy nhiên, với các dãy số lớn, hiệu suất của sắp xếp chèn cũng có thể không tốt.

Sắp xếp trộn (Merge Sort):

- Sắp xếp trộn cho hiệu suất tốt và ổn định, không phụ thuộc vào trạng thái ban đầu của dãy số.
- Nó phù hợp cho các dãy số lớn và là một lựa chọn tốt khi cần đảm bảo sắp xếp đúng đắn và ổn định.

Sắp xếp nhanh (Quick Sort):

- Sắp xếp nhanh cung cấp hiệu suất cao và thường là một lựa chọn tốt cho hầu hết các tình huống.
- Tuy nhiên, hiệu suất của nó có thể biến đổi dựa trên trạng thái ban đầu của dãy số và việc chọn pivot.

Tổng kết

- Đối với các dãy số nhỏ ($N = 100$), tất cả các thuật toán hoạt động tốt và không có sự khác biệt đáng kể về hiệu suất.

- Đối với các dãy số có kích thước trung bình ($N = 1000, 10000$), sắp xếp trộn và sắp xếp nhanh thường là lựa chọn tốt nhất vì chúng có hiệu suất ổn định và không quá nhạy cảm đối với trạng thái ban đầu của dãy số.
- Đối với các dãy số lớn ($N = 100000$), sắp xếp nhanh thường là thuật toán tốt nhất vì nó cung cấp hiệu suất cao và không phụ thuộc nhiều vào trạng thái ban đầu của dãy số.
- Chọn thuật toán sắp xếp phụ thuộc vào kích thước dãy số và đặc tính của dãy số. Nếu bạn có dãy số lớn hoặc không biết gì về trạng thái ban đầu của dãy số, sắp xếp nhanh và sắp xếp trộn thường là những lựa chọn an toàn và hiệu quả.