

Đề án giữa kì: CÁC THUẬT TOÁN SẮP XẾP

Môn Cấu Trúc Dữ Liệu & Giải Thuật

Lưu ý: Trong đề án này, ta xét bài toán sắp xếp tăng dần.

1. Yêu cầu của đề án

1.1. Tóm tắt sơ bộ

- Sinh viên sẽ cài đặt các thuật toán: Selection Sort, Insertion Sort, Bubble Sort, Heap Sort, Merge Sort, Quick Sort, Radix Sort
- Sau khi cài đặt xong các thuật toán ở trên, sinh viên sẽ tiến hành các thí nghiệm để có thể thấy được thời gian thực thi của mỗi thuật toán ứng với mỗi tình trạng dữ liệu đầu vào cũng như so sánh thời gian thực thi của các thuật toán với nhau. Ở đây, ta xét 4 tình trạng dữ liệu đầu vào:
 - Đã có thứ tự.
 - Có thứ tự ngược.
 - Ngẫu nhiên.
 - Gần như có thứ tự.

1.2. Cụ thể

Nội dung nộp bài của sinh viên bao gồm 2 phần: phần code (5đ) và phần báo cáo (5đ).

1.2.1. Phần code (5đ)

Sinh viên thí nghiệm với 4 tình trạng dữ liệu: đã có thứ tự (sorted), có thứ tự ngược (reversed), ngẫu nhiên (random), và gần như có thứ tự (nearly sorted). Với mỗi tình trạng, sinh viên khảo sát 5 kích thước dữ liệu (số phần tử) khác nhau: 1000, 3000, 10 000, 30 000, 100 000.

Với mỗi tình trạng dữ liệu **state**

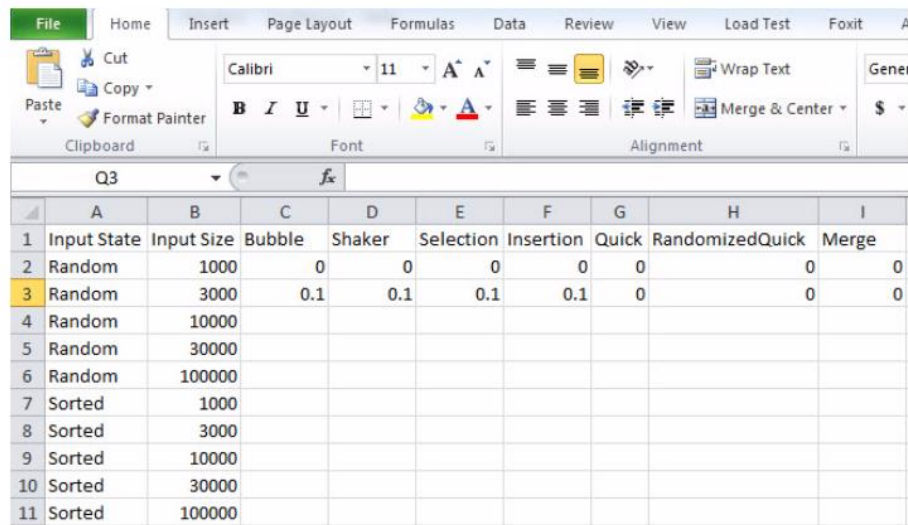
với mỗi kích thước dữ liệu **size**

với mỗi thuật toán sắp xếp **sortAlg**, sinh viên thực hiện các bước sau:

- Tạo mảng dữ liệu có tình trạng **state** và kích thước **size**
- Tiến hành sắp xếp mảng dữ liệu này bằng thuật toán **sortAlg** và đo thời gian thực thi của **sortAlg**
- Xuất kết quả sắp xếp ra file:
 - Mỗi phần tử cách nhau một khoảng trắng.
 - Tên file đặt theo định dạng **<sortAlg>_<state>_<size>.txt**Ví dụ: InsertionSort_NearlySorted_1000.txt

Các file kết quả khác: ngoài các file lưu kết quả sắp xếp như đã nói ở trên, để tiện lợi cho việc thống kê kết quả, chương trình của sinh viên cũng cần phải xuất thời gian thực thi của tất cả các lần chạy ở trên vào một file chung:

- Ta quy ước đặt tên file này là **Result.csv**. Việc sử dụng file ***.csv** có thể mở bằng Excel, thuận tiện cho việc vẽ đồ thị.
- File **Result.csv** sẽ có dạng như hình 1. Trong đó, ở mỗi dòng, ô đầu tiên là tình trạng dữ liệu ban đầu, ô thứ hai là kích thước dữ liệu, các ô tiếp theo là thời gian thực thi của các thuật toán.

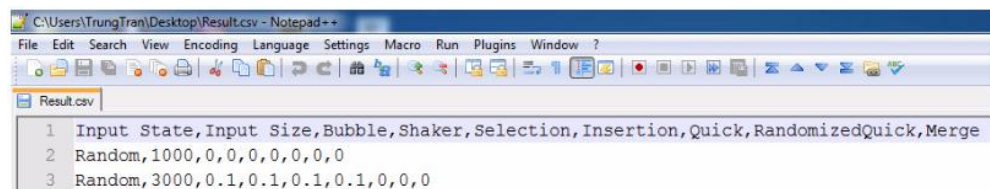


	A	B	C	D	E	F	G	H	I
1	Input State	Input Size	Bubble	Shaker	Selection	Insertion	Quick	RandomizedQuick	Merge
2	Random	1000	0	0	0	0	0	0	0
3	Random	3000	0.1	0.1	0.1	0.1	0	0	0
4	Random	10000							
5	Random	30000							
6	Random	100000							
7	Sorted	1000							
8	Sorted	3000							
9	Sorted	10000							
10	Sorted	30000							
11	Sorted	100000							

Hình 1. Minh họa file **Result.csv**

- Nếu mở file ***.csv** ở trên bằng Notepad++ thì sẽ thấy như hình 2. Như vậy, thực chất mỗi ô trên một dòng cách nhau bởi dấu phẩy (,). Ví dụ ta có thể ghi dòng thứ 2 bằng câu lệnh:

```
fprintf(f, "Random,1000,0,0,0,0,0,0,0\n");
```



```

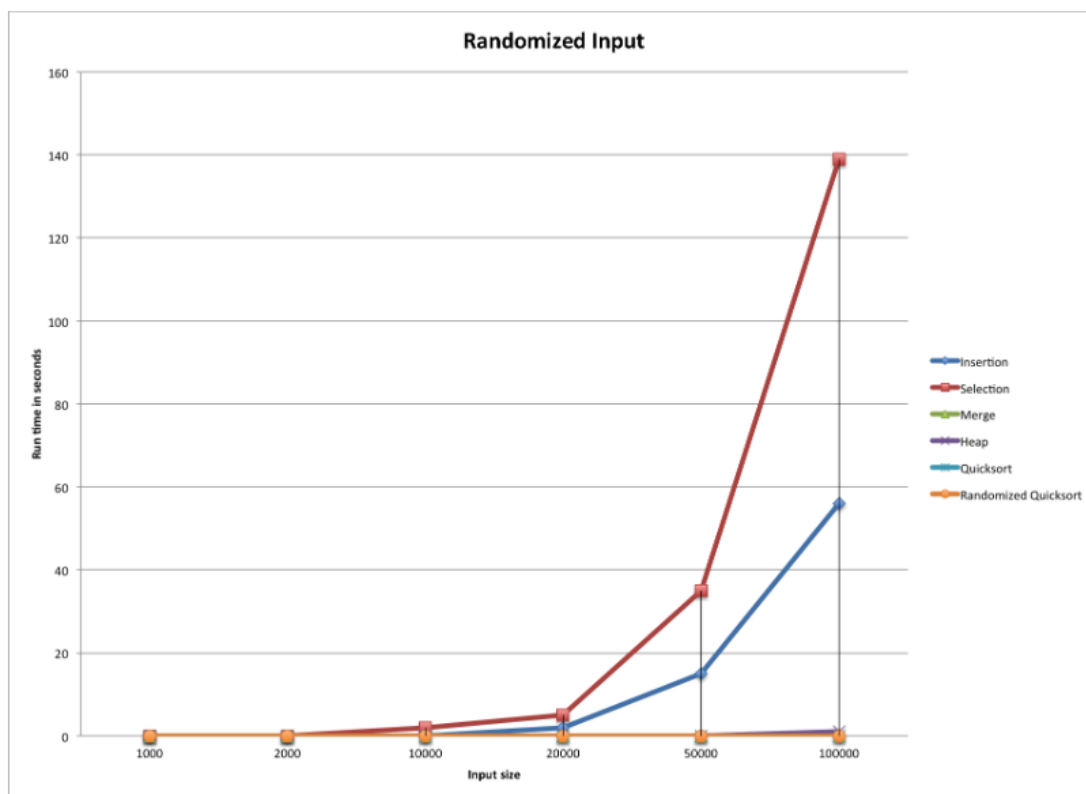
1 Input State,Input Size,Bubble,Shaker,Selection,Insertion,Quick,RandomizedQuick,Merge
2 Random,1000,0,0,0,0,0,0,0
3 Random,3000,0.1,0.1,0.1,0.1,0,0,0

```

Hình 2. File **Result.csv** mở bằng Notepad++

1.2.2. Phần báo cáo (5đ)

- Trình bày về các thuật toán sắp xếp ở trên (ý tưởng, thuật toán, độ phức tạp, đánh giá thuật toán). (2đ)
- Trình bày về kết quả thí nghiệm (2đ) và cho nhận xét (2đ). Cách trình bày kết quả thí nghiệm: thay vì đưa ra các con số về thời gian chạy, ta trực quan hóa chúng bằng đồ thị, như vậy, sẽ dễ quan sát và nhận xét hơn. Các bạn sẽ vẽ 4 đồ thị tương ứng với 4 tình trạng dữ liệu đầu vào. Trong đó, mỗi đồ thị có trục tung là kích thước dữ liệu, trục hoành là thời gian thực thi (có dạng như hình 3).



Hình 3. Minh họa dạng đồ thị

2. Một số quy định

- Cấu trúc thư mục MSSV gồm có:
 - Thư mục **Code**: chứa toàn bộ project (xóa thư mục debug, file *.ncb)
 - File **Report.doc(x)** hoặc **Report.pdf**
- Môi trường lập trình: Visual Studio 2012, 2013 hoặc 2015.
- Lưu ý: các trường hợp bài giống nhau sẽ bị 0 điểm.