

**Nhập môn phân tích độ phức tạp thuật toán – CQ 2023.****BÀI TẬP THỰC HÀNH LẦN 2****Câu 1. (4 điểm)**

Cài đặt bổ sung các biến đếm số phép gán và số phép so sánh vào mã nguồn của thuật toán sau đây, thực hiện đếm rồi lập bảng thống kê lại với tất cả giá trị  $n = 1, 2, 3, \dots, 40$  cho thuật toán tính số Fibonacci không đệ quy (SV có thể cài đặt lại bằng Python cũng được):

```

1 int fibonacci(int n)
2 {
3     if (n <= 1)
4         return 1;
5
6     int last = 1;
7     int nextToLast = 1;
8     int answer = 1;
9
10    for (int i = 2; i <= n; i++)
11    {
12        answer = last + nextToLast;
13        nextToLast = last;
14        last = answer;
15    }
16
17    return answer;
18 }

```

**Câu 2. (6 điểm)**

Xét bài toán: “cho hai chuỗi  $s_1, s_2$  gồm các ký tự tiếng Anh in hoa và có cùng độ dài  $n$  không vượt quá  $10^5$ . Kiểm tra xem chuỗi này có thể nhận được từ chuỗi kia bằng cách sắp xếp lại một cách thích hợp thứ tự các ký tự trong chuỗi kia hay không?”. Có hai ý tưởng như sau:

- Duyệt qua từng ký tự của  $s_1$ , kiểm tra xem ký tự đó có nằm trong  $s_2$  không rồi đánh dấu lại. Nếu như cả  $n$  ký tự của  $s_2$  đều được đánh dấu thì trả về *true*, ngược lại là *false*.
- Dùng các biến đếm số lần xuất hiện của các ký tự ‘A’, ‘B’, ..., ‘Z’ có trong hai chuỗi  $s_1, s_2$  và xem chúng có bằng nhau không. Nếu bằng hết thì trả về *true*, ngược lại là *false*.

Hãy thực hiện các yêu cầu bên dưới:

- Xây dựng chương trình (dùng C++/Python) theo các hướng trên để giải quyết bài toán.
- Sử dụng phương pháp thực nghiệm để khảo sát thời gian chạy của từng thuật toán trên: kiểm tra với 5 mốc giá trị của  $n$  thích hợp, ứng với mỗi giá trị thì sinh ngẫu nhiên ra vài cặp chuỗi  $s_1, s_2$  để test và tính giá trị trung bình thời gian chạy. Vẽ biểu đồ mô tả.
- Dựa trên phân tích ở b), hãy đánh giá về thời gian thực thi và phân tích sơ lược về tính trade-off giữa không gian lưu trữ bổ sung & thời gian thực thi.
- Ngoài hai ý tưởng i) và ii) ở trên, anh/chị còn có ý tưởng nào khác để giải bài này không?

**HẾT**