

Phân Tích Thuật Toán

Bùi Thị Thanh Phương

Ngày 29 tháng 5 năm 2023

Bài toán Ta xây dựng một dãy các từ Fibonacci như sau:

$$\begin{cases} f_0 = "abc", f_1 = "def" \\ f_{n+1} = \text{strcat}(f_n, f_{n-1}), \forall n \geq 1 \end{cases}$$

Trong đó, hàm *strcat* là hàm nối hai chuỗi thành một chuỗi mới.

Ví dụ: *strcat*("abc", "def") = "abcdef"

- (a) Hãy xây dựng thuật toán để tìm chính xác ký tự thứ k của từ $f_n, \forall n \geq 0$
- (b) Đánh giá độ phức tạp của thuật toán đưa ra.

Lời giải

Việc xây dựng thuật toán tạo dãy các từ Fibonacci tương tự với việc xây dựng dãy số Fibonacci. Do với ngôn ngữ Python, nối 2 chuỗi tương tự với việc dùng phép cộng (+) giữa 2 biến. Khi đó ta xây dựng thuật toán tìm chính xác ký tự thứ k của từ f_n như sau

Function fibWord(n, k):

```
f0 ← "abc";
f1 ← "def";
result ← "";
if  $n == 0$  then
    | result ← f0;
end
else if  $n == 1$  then
    | result ← f1;
end
else
    | for  $i = 2$  to  $n$  do
        | result ← f1;
        | f1 ← f1 + f0;
        | f0 ← result;
    | end
end
return result, result[k - 1];
```

Thực thi thuật toán trên ta thu được kết quả như sau

Với cách tiếp cận này thì thuật toán trên có độ phức tạp là $O(N)$. Ta có thể kiểm chứng qua biểu đồ sau Ta thấy rằng, đường biểu diễn số phép so sánh song song với đường $O(N)$, và đường biểu diễn số phép gán có chênh lệch, nhưng vẫn có dạng song song với đường $O(N)$. Chứng tỏ rằng, thuật toán trên có độ phức tạp $O(N)$.

5th fibonacci string: defabcdefdefabc
5th char of 5th fibonacci string: b

