

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



Nhập môn phân tích độ phức tạp thuật toán
Bài đọc: Introduction to the Design and Analysis of Algorithms - Chapter 1

GVLT: Thầy Trần Đan Thư

GVTH: Thầy Nguyễn Đức Thân
Thầy Trương Toàn Thịnh
Thầy Nguyễn Vinh Tiệp
Thầy Nguyễn Sơn Hoàng Quốc

Sv: Nguyễn Phan Mạnh Hùng 1312727

1 Câu hỏi tìm hiểu

1.1 Câu 1

Câu hỏi: Hãy nêu khái niệm độ phức tạp về thời gian và độ phức tạp về không gian.

Trả lời:

- Độ phức tạp thời gian thể hiện đơn vị thời gian cần thiết (thể hiện tốc độ) của thuật toán để xử lý.
- Độ phức tạp không gian thể hiện đơn vị không gian cần thiết, bên cạnh không gian lưu trữ input và output, để thuật toán xử lý.

1.2 Câu 2

Câu hỏi: Khái niệm kích thước đầu vào bài toán (input's size) là gì? Hãy nêu ví dụ một bài toán mà có thể chọn nhiều hơn 1 kích thước đầu vào.

Trả lời:

- Input's size là kích thước dữ liệu đầu vào của bài toán. Khái niệm này nên đi liền với từng loại bài toán và thuật toán cụ thể, do có thể với cùng 1 bộ dữ liệu, thì mỗi loại bài toán khác nhau sẽ quan tâm tới các thành phần khác nhau của bộ dữ liệu đó, điều này tương tự với từng thuật toán khác nhau.
- Ta xét bài toán đếm số thành phần liên thông của đồ thị. Để giải quyết bài toán này ta thực hiện thuật toán loang (bfs, dfs,...). Tuy vậy với cách thể hiện đồ thị khác nhau thì ta quan tâm tới các kích thước khác nhau. Ví dụ nếu ta chọn tổ chức đồ thị theo ma trận kề thì ta sẽ quan tâm tới số đỉnh N , và độ phức tạp tính toán $\in O(N^2)$. Còn nếu tổ chức theo danh sách kề thì độ phức tạp $\in O(M)$, với M là số lượng cạnh.

1.3 Câu 3

Câu hỏi: Khái niệm thao tác cơ bản (basic operation) là gì và có ý nghĩa như thế nào trong việc đo độ phức tạp về thời gian?

Trả lời: Basic operation là những operation quan trọng nhất của thuật toán, nghĩa là những operation chiếm phần lớn thời gian thực hiện trong tổng thời gian tính toán của thuật toán. Basic operation đóng vai trò vô cùng quan trọng vì khi dữ liệu đủ lớn thì thời gian tính toán các basis operations sẽ xấp xỉ thời gian chạy của thuật toán.

1.4 Câu 4

Câu hỏi: Tại sao người ta quan tâm đến khái niệm bậc tăng trưởng của độ phức tạp (Orders of Growth)?

Trả lời: Do sự khác biệt về mặt thời gian (trên dữ liệu nhỏ) khó có thể dùng để phân biệt được giữa các thuật toán hiệu quả và không hiệu quả. Chưa kể, trên thực tế đối với nhiều bài toán thì kích thước dữ liệu phải rất lớn để có thể thấy rõ sự khác biệt giữa các thuật toán, tuy vậy nguồn tài nguyên của máy tính không đáp ứng được. Do đó, ta nên dùng bậc tăng trưởng của độ phức tạp để đánh giá, vì nó cho ta thấy khá trực quan mối quan hệ giữa Input's size và thời gian tính toán.

1.5 Câu 5

Câu hỏi: Ý nghĩa của độ phức tạp trong trường hợp tốt nhất của thuật toán là gì?

Trả lời:

- Khái niệm: là hiệu năng tốt nhất của thuật toán đối với trường hợp tốt nhất của bộ dữ liệu theo kích thước N .
- Ta có thể tận dụng điều này bởi vài thuật toán cũng thể hiện tốt trên các bộ dữ liệu gần giống với bộ dữ liệu ở trường hợp tốt nhất. Ví dụ, ta thấy thuật toán insertion sort thể hiện tốt nhất trên dãy đã sắp xếp. Tuy vậy đối với các dãy đã được sắp xếp gần hết (almost-sorted array) thì thuật toán này chạy cũng khá tốt.

1.6 Câu 6

Câu hỏi: Vai trò của luật L'Hôpital trong so sánh độ phức tạp của 2 thuật toán là gì? Nêu ví dụ.

Trả lời:

- Đây là một công cụ hiệu quả để so sánh độ phức tạp của hai thuật toán trong một vài trường hợp so với cách tính thông thường, ví dụ như bằng định nghĩa.

- Ví dụ ta so sánh $f(n) = \ln(n)$ và $g(n) = n^2$.

$$\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} = \lim_{n \rightarrow \infty} \frac{g'(n)}{f'(n)} = \lim_{n \rightarrow \infty} \frac{2n}{\frac{1}{n}} = \lim_{n \rightarrow \infty} 2n^2 = \infty \quad (1)$$

Vậy $g(n)$ có bậc tăng trưởng lớn hơn $f(n)$.