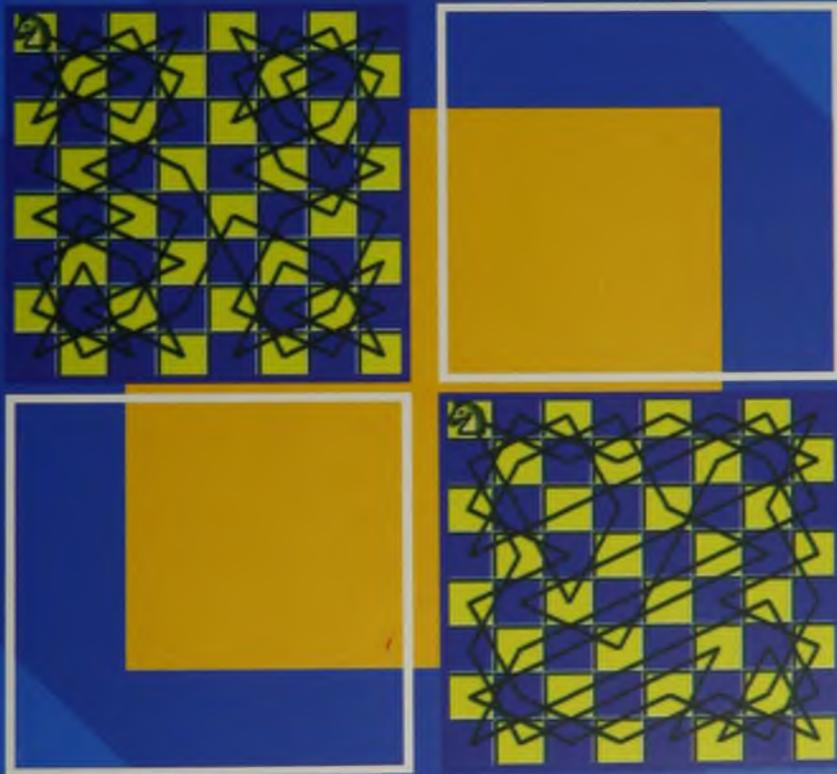


3  
TRẦN THÔNG QUẾ

# CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

(PHÂN TÍCH VÀ CÀI ĐẶT TRÊN C/C++)

(Tập 1)



**ic**  
PUBLISHER

NHÀ XUẤT BẢN THÔNG TIN VÀ TRUYỀN THÔNG

TRẦN THÔNG QUẾ

# CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

(PHÂN TÍCH VÀ CÀI ĐẶT TRÊN C/C++)

(Tập 1)

NHÀ XUẤT BẢN THÔNG TIN VÀ TRUYỀN THÔNG

**Mã số: HT 04 HM14**

## LỜI NHÀ XUẤT BẢN

Trong thời đại bùng nổ công nghệ thông tin như hiện nay, dữ liệu đang được sinh ra ngày càng nhiều và dưới nhiều hình thức khác nhau. Việc xử lý nhằm ứng dụng kho dữ liệu quý giá đó như thế nào để đạt được hiệu quả cao nhất luôn là vấn đề được rất nhiều những chuyên gia về công nghệ thông tin đặc biệt quan tâm. Chính vì lý do đó mà *Cấu trúc dữ liệu và thuật toán* vẫn luôn là một chủ đề mà ngày càng được đầu tư nghiên cứu và phát triển, nó không bị lỗi thời trong bất cứ xu hướng nào của lĩnh vực Công nghệ thông tin, việc phát triển về thuật toán ngày một hoàn thiện hơn gần như chưa bao giờ dừng lại và không có giới hạn.

Trước nhu cầu thiết thực cần tìm hiểu của bạn đọc, tác giả Trần Thông Quế - giảng viên cao cấp đã có trên 35 năm kinh nghiệm giảng dạy ở lĩnh vực Công nghệ thông tin và luôn tâm huyết với chủ đề này đã phối hợp với Nhà xuất bản Thông tin và Truyền thông xuất bản bộ sách "*Cấu trúc dữ liệu và thuật toán - phân tích và cài đặt trên C/C++*", gồm 2 tập.

Tập 1 gồm 5 chương, cụ thể như sau:

*Chương 1: Các khái niệm cơ bản về thuật toán và cấu trúc dữ liệu*

*Chương 2: Các thuật toán tìm kiếm trong và sắp xếp trong*

*Chương 3: Một số chiến lược thiết kế thuật toán*

*Chương 4: Các kiểu dữ liệu trừu tượng và biến nhớ động*

*Chương 5: Cấu trúc cây*

Với kinh nghiệm nhiều năm giảng dạy, tác giả đã lựa chọn cách tiếp cận đơn giản hóa các vấn đề phức tạp, từ cơ bản đến nâng cao.

Nội dung cuốn sách được trình bày một cách chi tiết, dễ hiểu, sau mỗi nội dung đều có ví dụ minh họa, sau mỗi thuật toán đều được đánh giá độ phức tạp và cài đặt demo. Cuốn sách còn bổ sung thêm một số thuật toán mà những cuốn sách đã xuất bản trước đây chưa có như: Thuật toán sắp xếp rung lắc, thuật toán sắp xếp với độ dài bước giảm dần... Chính vì vậy, cuốn sách này hướng tới đối tượng độc giả đông đảo, từ những bạn đọc mới làm quen tới những bạn đọc đã có kinh nghiệm nhiều năm trong lĩnh vực này. Đây thực sự là tài liệu bổ ích dành cho sinh viên, giảng viên, lập trình viên cũng như các chuyên gia về công nghệ thông tin có nhu cầu tìm hiểu và nghiên cứu.

Xin trân trọng giới thiệu cùng bạn đọc./.

**NHÀ XUẤT BẢN THÔNG TIN VÀ TRUYỀN THÔNG**

## LỜI NÓI ĐẦU

Cấu trúc dữ liệu và Thuật toán (Data Structure and Algorithms) là môn học bắt buộc không những với mỗi sinh viên ngành Công nghệ Thông tin mà còn là môn học và môn thi quốc gia đầu vào bắt buộc với các nghiên cứu viên của ngành học đó. Nó là một trong các môn học khó của ngành CNTT, đặc biệt càng khó đối với hầu hết các sinh viên khi phải cài đặt một thuật toán hay một bài toán nào đó thuộc môn học này.

Dựa trên thực tế đó, bằng kinh nghiệm tích lũy được trong nhiều năm liên tục giảng dạy lý thuyết và thực hành môn học “*Data Structure and Algorithms*”, chúng tôi biên soạn cuốn sách này nhằm đáp ứng kiến thức một cách tối ưu cho mỗi sinh viên, các nghiên cứu viên ngành CNTT và cả các giảng viên phụ trách bộ môn này với thâm niên giảng dạy còn chưa nhiều.

Bất cứ sự thành công nào của một dự án Tin học đều là kết quả của việc kết hợp khéo léo giữa Cấu trúc dữ liệu và Thuật toán. Khẳng định này được chứng tỏ trong một công thức rất ngắn gọn mang tính triết lý đương đại nghề nghiệp:

**BIG DATA + BIG COMMUNITY = BIG RESULT**

**Dữ liệu lớn + Kết hợp lớn = Kết quả lớn**

Việc cài đặt các thuật toán cơ bản hoặc nổi tiếng hoặc khó được thực hiện trên các ngôn ngữ chuyên nghiệp đương đại là C/C++ nhằm giải đáp câu hỏi tồn tại trong đầu những người học là: Thực thi các thuật toán ấy trên máy tính điện tử như thế nào và sẽ cho kết quả ra sao?

Chúng tôi tin tưởng rằng tài liệu này sẽ rất bổ ích và hữu hiệu cho tất cả những ai có nhu cầu tìm hiểu về cấu trúc dữ liệu và giải thuật - một môn học tuy phức tạp song không kém phần lý thú, hấp dẫn.

Mặc dù đã có nhiều năm giảng dạy môn học này ở không ít trường đại học công và tư lập, song do những hạn chế chủ quan của bản thân nên cuốn sách sẽ khó tránh khỏi những khiếm khuyết, rất mong các độc giả xa gần cho các ý kiến “phản biện” để các lần tái bản sau sẽ ngày càng hoàn chỉnh hơn.

Xin chân thành cảm ơn các độc giả!

**Tác giả**  
**TRẦN THÔNG QUẾ**

## Chương 1

# CÁC KHÁI NIỆM CHUNG VỀ THUẬT TOÁN VÀ CẤU TRÚC DỮ LIỆU

### 1.1. MỘT VÀI KHÁI NIỆM VỀ THUẬT TOÁN

Về thuật ngữ, tiếng Anh “thuật toán” là “algorithm” có xuất xứ từ tên của nhà toán học cổ ở Trung Á là Abu Ja'fa Mohammed Ibn Musa Al Khowarizmi (khoảng năm 825). Trong khi nghiên cứu cách giải một loạt các bài toán đại số, Al Khowarizmi nhận thấy có một lớp những bài toán cùng kiểu có chung một cách giải. Và ông đã quy trình hóa cách giải lớp các bài toán đó bằng một tập hợp các bước giải có *trình tự trước sau* mà trước ông, các nhà toán học cổ đại chưa làm. Để kỷ niệm thành quả này của ông, các “hậu duệ” của ông đã lấy tên ông đặt cho những quy trình giải các bài toán cùng kiểu ấy là Al Khowarizmi. Dần dần theo thời gian, các nhà toán học đã *Anh hóa* thuật ngữ này thành **Algorithm** hiện đại, ngắn gọn và tiện dùng như ngày nay.

**1.1.1. Định nghĩa thuật toán:** Một cách đơn giản và dễ hiểu, song thiếu chặt chẽ, thiếu chính xác, khi hiểu thuật toán là cách giải bài toán. Về thuật ngữ này có *khá nhiều tác giả* đã phát biểu khác nhau, thể hiện các cách tiếp cận khác nhau về cùng một khái niệm. Trong các cách định nghĩa về thuật toán, định nghĩa sau đây ngắn gọn nhất, rõ ràng nhất và chặt chẽ nhất:

*“Thuật toán là một tập hợp hữu hạn các chỉ thị (Instructions) hoặc các thao tác (Operations) được sắp xếp theo một trật tự xác định để dạy cho con người, máy tính hoặc robot giải một bài toán hoặc giải quyết một vấn đề nào đó”.*

Từ định nghĩa này, ta trực tiếp suy ra các tính chất dưới đây của thuật toán:

#### 1.1.2. Các tính chất của thuật toán

**a. Tính hữu hạn (tính dừng):** Bất kỳ thuật toán nào cũng phải dừng sau một số hữu hạn bước. Tức là mọi thuật toán đều phải có **Bắt đầu** và **Kết thúc**. (Cuốn sách này không bàn về những thuật toán có vô hạn các chỉ thị (một ví dụ dễ thấy nhất là các thuật toán chạy lặp vòng (quần) vô kỳ hạn, tức nó không dừng). Có nhiều sinh viên mới học lập trình trên một ngôn ngữ bậc cao nào đó, khi thực hành rất hay mắc lỗi quên tính chất này. Cụ thể hơn: các sinh viên đó

quen viết từ khóa bắt đầu (trong  $C/C^{++}$  là ký tự  $\{$ ) hoặc từ khóa kết thúc (trong  $C/C^{++}$  là ký tự  $\}$ ) của ngôn ngữ đang thực hành cho chương trình đó, dấu rằng trình dịch đã bắt lỗi mà cứ lóng ngóng mãi không sửa được lỗi cơ bản này. Hãy luôn thường trực trong đầu tính chất này của thuật toán khi lập trình.

**b. Tính xác định đơn nhất:** Tại mỗi bước của thuật toán, các chỉ thị phải rõ ràng, chuẩn xác. Các chỉ thị *không được diễn đạt bằng những ngôn từ mập mờ, hình như*, hoặc có thể hiểu theo những nghĩa khác nhau. Sự diễn đạt các chỉ thị *chỉ có một cách hiểu duy nhất*. Nói rõ hơn, trong cùng một điều kiện, **những người khác nhau, những máy tính khác nhau** cùng thực thi một chỉ thị nào đó của thuật toán thì phải **cho cùng một kết quả**.

**c. Tính đúng đắn:** Sau khi thực thi xong toàn bộ các bước của thuật toán, ta phải thu được kết quả mong muốn. Kết quả này phải được người xây dựng thuật toán xác định trước trong lúc xây dựng thuật toán.

**d. Tính phổ dụng:** Thuật toán phải giải được tất cả các bài toán thuộc cùng một lớp các bài toán cùng kiểu.

*Ví dụ 1.1:* Dân số Việt Nam hiện tại là 86 triệu người. Tỷ lệ bình quân tăng dân số hàng năm là 0,003%.

Tính xem sớm nhất đến năm nào dân số nước ta đạt ít nhất 100 triệu người?

*Ví dụ 1.2:* Tôi hiện có số vốn là 130 triệu đồng. Lãi suất hàng năm thu được là 8%. Tính xem sớm nhất đến năm nào tôi thu được số tiền (cả vốn lẫn lãi) là 300 triệu đồng?

Thuật toán xây dựng cho bài toán ở ví dụ 1.1 cũng phải dùng được cho bài toán ở ví dụ 1.2 và ngược lại (đó là hai bài toán cùng kiểu, dấu rằng ý nghĩa của dữ liệu hoàn toàn khác nhau).

**e. Tính hiệu quả:** Tính chất này của thuật toán được đánh giá trên các chỉ tiêu sau:

- Thuật toán sử dụng tiết kiệm nhất các tài nguyên (resources) của máy tính như dung lượng bộ nhớ trong, bộ nhớ ngoài.

- Thực thi trên máy tính (chạy) nhanh nhất có thể được. (Chỉ tiêu này có tên khác là: thời gian chạy chương trình hoặc thời gian thực hiện thuật toán). Về mặt lý thuyết, chỉ tiêu này đáng chú ý nhất.

- Dễ cài đặt (Set Up) trên máy tính.

### 1.1.3. Ba đặc trưng của thuật toán

Bất kỳ thuật toán nào cũng có ba đặc trưng sau:

a. *Vào dữ liệu (Data Input)*. Hiển nhiên là bất kỳ thuật toán nào cũng phải có bộ dữ liệu vào. Đó là tập các dữ liệu phải nạp vào thuật toán khi nó bắt đầu được thực thi.

b. *Xử lý dữ liệu (Data Process)*. Mọi thao tác trên dữ liệu vào (nói riêng là tính toán) ta gọi là xử lý chúng để đạt được kết quả của bài toán. Công việc đặc trưng này về thực chất là nội dung chủ yếu (chính là các bước thực thi) của thuật toán. Không có đặc trưng này thì bất thành thuật toán.

c. *Xuất dữ liệu (Data Output)*. Cuối cùng thì thuật toán cũng phải cho ra kết quả giải bài toán hay giải quyết vấn đề. Không thể khác được! Việc này trong Tin học gọi là xuất dữ liệu. Các dữ liệu được xuất này có liên quan tới bộ dữ liệu vào và *quá trình “chế biến” nó bởi các bước của thuật toán* (công đoạn *Data Process*), đây là kết quả cần có sau khi chạy (thực thi) thuật toán.

### 1.1.4. Các cách diễn đạt thuật toán (Expressing Algorithms)

Dưới đây là những cách hay dùng để diễn đạt thuật toán.

a. **Diễn đạt thuật toán bằng ngôn ngữ tự nhiên**. Ngôn ngữ tự nhiên là ngôn ngữ mà con người của một quốc gia bất kỳ hàng ngày dùng để làm việc và giao tiếp. Ưu điểm của phương pháp này là dễ viết (vì ngôn ngữ tự nhiên quen thuộc từ nhỏ với mọi người). Nhược điểm độc nhất là máy tính *“không hiểu”* được thuật toán diễn đạt theo cách này. Do đó ta không thể chuyển giao thuật toán diễn đạt bằng ngôn ngữ tự nhiên cho máy tính.

*Ví dụ 1.3:* Diễn đạt bằng ngôn ngữ tự nhiên thuật toán tìm ước số chung lớn nhất (USCLN) của hai số nguyên dương tùy ý nhờ phương pháp trừ liên tiếp của Euclide.

Bước 1: (**Bắt đầu**) Nhập 2 số nguyên dương bất kỳ (vào thuật toán);

Bước 2: Chừng nào mà  $a-b \neq 0$  (tức  $a \neq b$ ) thì lặp các thao tác sau:

- Nếu  $a=b$  thì USCLN= $a$ ; Đến bước 3;

- Nếu  $a \neq b$  thì:

+ Nếu  $a>b$  thì USCLN= $a-b$ . Quay lại bước 2.

+ Ngược lại ( $a<b$ ) thì USCLN( $a,b$ )= $b-a$ . Quay lại bước 2;

Bước 3: (**Kết thúc**) Xuất kết quả.