

SỞ GIÁO DỤC VÀ ĐÀO TẠO HÀ NỘI

NGUYỄN QUANG UY

# GIÁO TRÌNH JAVA

*(Dùng trong các trường THCN)*



NHÀ XUẤT BẢN HÀ NỘI - 2007



## Lời giới thiệu

---

**N**ước ta đang bước vào thời kỳ công nghiệp hóa, hiện đại hóa nhằm đưa Việt Nam trở thành nước công nghiệp văn minh, hiện đại.

Trong sự nghiệp cách mạng to lớn đó, công tác đào tạo nhân lực luôn giữ vai trò quan trọng. Báo cáo Chính trị của Ban Chấp hành Trung ương Đảng Cộng sản Việt Nam tại Đại hội Đảng toàn quốc lần thứ IX đã chỉ rõ: “Phát triển giáo dục và đào tạo là một trong những động lực quan trọng thúc đẩy sự nghiệp công nghiệp hóa, hiện đại hóa, là điều kiện để phát triển nguồn lực con người - yếu tố cơ bản để phát triển xã hội, tăng trưởng kinh tế nhanh và bền vững”.

Quán triệt chủ trương, Nghị quyết của Đảng và Nhà nước và nhận thức đúng đắn về tầm quan trọng của chương trình, giáo trình đối với việc nâng cao chất lượng đào tạo, theo đề nghị của Sở Giáo dục và Đào tạo Hà Nội, ngày 23/9/2003, Ủy ban nhân dân thành phố Hà Nội đã ra Quyết định số 5620/QĐ-UB cho phép Sở Giáo dục và Đào tạo thực hiện đề án biên soạn chương trình, giáo trình trong các trường Trung học chuyên nghiệp (THCN) Hà Nội. Quyết định này thể hiện sự quan tâm sâu sắc của Thành ủy, UBND thành phố trong việc nâng cao chất lượng đào tạo và phát triển nguồn nhân lực Thủ đô.

Trên cơ sở chương trình khung của Bộ Giáo dục và Đào tạo ban hành và những kinh nghiệm rút ra từ thực tế đào tạo, Sở Giáo dục và Đào tạo đã chỉ đạo các trường THCN tổ chức biên soạn chương trình, giáo trình một cách khoa học, hệ

thống và cập nhật những kiến thức thực tiễn phù hợp với đời sống học sinh THCS Hà Nội.

Bộ giáo trình này là tài liệu giảng dạy và học tập trong các trường THCS ở Hà Nội, đồng thời là tài liệu tham khảo hữu ích cho các trường có đào tạo các ngành kỹ thuật - nghiệp vụ và đông đảo bạn đọc quan tâm đến vấn đề hướng nghiệp, dạy nghề.

Việc tổ chức biên soạn bộ chương trình, giáo trình này là một trong nhiều hoạt động thiết thực của ngành giáo dục và đào tạo Thủ đô để kỷ niệm "50 năm giải phóng Thủ đô", "50 năm thành lập ngành" và hướng tới kỷ niệm "1000 năm Thăng Long - Hà Nội".

Sở Giáo dục và Đào tạo Hà Nội chân thành cảm ơn Thành ủy, UBND, các sở, ban, ngành của Thành phố, Vụ Giáo dục chuyên nghiệp Bộ Giáo dục và Đào tạo, các nhà khoa học, các chuyên gia đầu ngành, các giảng viên, các nhà quản lý, các nhà doanh nghiệp đã tạo điều kiện giúp đỡ, đóng góp ý kiến, tham gia Hội đồng phản biện, Hội đồng thẩm định và Hội đồng nghiệm thu các chương trình, giáo trình.

Đây là lần đầu tiên Sở Giáo dục và Đào tạo Hà Nội tổ chức biên soạn chương trình, giáo trình. Dù đã hết sức cố gắng nhưng chắc chắn không tránh khỏi thiếu sót, bất cập. Chúng tôi mong nhận được những ý kiến đóng góp của bạn đọc để từng bước hoàn thiện bộ giáo trình trong các lần tái bản sau.

GIÁM ĐỐC SỞ GIÁO DỤC VÀ ĐÀO TẠO

## Lời nói đầu

---

*Trong thời đại công nghệ thông tin bùng nổ như ngày nay, việc lập được các chương trình chạy trên máy tính nhằm thực hiện các công việc nào đó là điều vô cùng cần thiết. Tuy nhiên, xung quanh chúng ta có quá nhiều thứ ngôn ngữ lập trình: Pascal, C, C++, Basic, Delphi, Cobol,... Vậy đâu là ngôn ngữ lập trình phù hợp với bạn? Câu trả lời trước tiên phụ thuộc vào chính bạn.*

*Nếu bạn là người mong muốn trở thành một lập trình viên chuyên nghiệp; nếu bạn là người mong muốn xây dựng các chương trình hướng đối tượng; nếu bạn phải thường xuyên xây dựng và vận hành các ứng dụng chạy trên mạng hay Internet; nếu bạn là người thường xuyên giải quyết các bài toán quản lý;... thì Java là ngôn ngữ lập trình hữu ích giúp bạn đạt kết quả cao.*

*Nhiều người cho rằng, Java là một bước cách tân vĩ đại của thế giới tin học. Nếu bạn đã từng thử thách sức mình với một ngôn ngữ lập trình khác như C, C++, Pascal... thì với Java bạn sẽ thấy một cách suy nghĩ hoàn toàn khác, một con đường mới mẻ để đi đến mục đích của bạn.*

*Tuy nhiên, nếu bạn chưa từng bao giờ biết đến lập trình, chưa có một khái niệm nào về ngôn ngữ lập trình thì cuốn sách này vẫn là để dành cho bạn, nhưng để đi đến đích, bạn phải làm việc nhiều và rất nhiều.*

*Cũng cần lưu ý thêm, mặc dù Java là một ngôn ngữ lập trình mạng vô cùng mạnh mẽ nhưng đây là một cuốn sách thiên về lập trình hướng đối tượng trong Java. Vì vậy, phần lập trình mạng sẽ không được đề cập đến ở đây. Các bạn độc giả quan tâm đến lĩnh vực đó có thể tìm đọc trong một số tài liệu khác.*

TÁC GIẢ



# Chương 1

## TỔNG QUAN VỀ JAVA

### I. LỊCH SỬ PHÁT TRIỂN JAVA

#### 1. Khởi điểm

Cuối năm 1990, James Gosling được công ty Sun Microsystems giao nhiệm vụ xây dựng một phần mềm lập trình cho các máy hàng điện dân dụng. Lúc đầu Gosling và nhóm cộng sự định lựa chọn C++ nhưng nhanh chóng nhận thấy rằng C++ không thích hợp vì quá cồng kềnh và đòi hỏi rất nhiều tài nguyên (bộ nhớ, đĩa...). Gosling đã quyết định xây dựng hẳn một ngôn ngữ lập trình mới và đặt tên là Oak (Oak: cây sồi, vì trước cửa phòng làm việc của ông nhìn ra một cây sồi).

Đầu tiên, Oak được sử dụng trong “Dự án Xanh” (Green Project) trong đó Java được sử dụng để kiểm soát các thiết bị điện dân dụng trong dự án. Sau đó, Oak được sử dụng trong dự án Phim-theo-yêu cầu (Video - on - demand Project). Qua các dự án đó, Java được phát triển nhanh chóng, đến lúc này Oak được đổi tên vì có sự trùng lặp và Java ra đời.

#### 2. Sự phát triển

Java được sử dụng chủ yếu trong bộ công cụ phát triển Java (*Java Development Kit - JDK*) như là một thư viện chuẩn, trong đó chứa trình biên dịch, thông dịch, trợ giúp, soạn tài liệu,...

Khởi đầu với JDK 1.0 vào năm 1995, JDK 1.1 được công bố vào năm 1996 với nhiều cải tiến như: tăng thêm các hàm giao diện (AWT), xây dựng hệ thống thư viện dùng lại Javabeans, JFC,... Bộ Java 1.6 là lõi cho việc viết các IED (*Integrated Environment Development*) nổi tiếng như *Jbuilder 2.0*, *Visual J++ 6.0*,... Trong giáo trình này chúng ta yêu cầu các bản JDK từ 1.2 trở về sau.

JDK 1.2 ra đời năm 1998 và được coi như một bước ngoặt trong quá trình phát triển của Java. JDK 1.2 chứa rất nhiều những thay đổi trong cấu trúc so với JDK 1.1. Khả năng bảo mật cao hơn và linh động hơn, nâng cao tính khả chuyển và tốc độ thực hiện, một chương trình được viết bằng JDK 1.2 chạy nhanh hơn 30% đến 40% so với một chương trình viết bằng JDK 1.1. Đặc biệt, JDK còn phát triển thêm JDK đồ họa 2D và 3D nâng cao đáng kể khả năng đồ họa của Java.

Sau Java 1.2, cho đến ngày nay, Java đã phát triển đến phiên bản 1.4 và với tốc độ như vậy có lẽ là sau vài tháng nữa chúng ta lại được nghe đến những thông tin mới về Java.

## II. CÁC ĐẶC ĐIỂM CỦA JAVA

Có nhiều đặc điểm nổi bật của Java so với các ngôn ngữ khác. Dưới đây là một số đặc điểm chủ yếu nhất.

### 1. Đơn giản (Simple)

Đặc điểm đầu tiên và cũng là mục đích của Java là tính đơn giản. Mục đích của những người sáng lập ra Java là để thay thế cho C và C++. Mặc dù công việc này chẳng hề đơn giản chút nào nhưng ta cũng cần thấy rằng Java đã hạn chế được rất nhiều tính phức tạp ở trong C và C++. Hai ví dụ có thể chứng minh cho tính đơn giản của Java so với C và C++ đó là Java đã bỏ đi khái niệm con trỏ ở trong C và bỏ đi tính đa thừa kế ở trong C++.

Một điều nữa có thể minh chứng cho tính đơn giản của Java là kích thước của bộ biên dịch cơ bản và lớp hỗ trợ là rất nhỏ. Bản JDK 1.2 chỉ chiếm dung lượng 4 Mb so với vài trăm Mb của *Visual C++* và *Visual Basic*.

### 2. Hướng đối tượng

Java là một ngôn ngữ lập trình thuần túy hướng đối tượng. Có nghĩa là trong Java không có bất cứ một biến hay một thủ tục nào được viết ở ngoài lớp. Mọi ứng dụng viết trên Java đều phải được xây dựng từ các đối tượng và thông qua các đối tượng. Nếu như trong C/C++ ta có thể tạo ra các hàm toàn cục (*global function*), thì ở Java chúng ta chỉ có thể tạo ra các hàm tương tự thông qua một đối tượng nào đó.

### 3. Phân tán

Java được thiết kế để hỗ trợ các ứng dụng phân tán. Ứng dụng phân tán là những ứng dụng làm việc trên môi trường mạng. Trong Java để xây dựng các



ứng dụng như thế ta sử dụng lớp mạng (*java.net*). Ví dụ với lớp URL của Java - một ứng dụng Java có thể dễ dàng truy xuất đến một máy chủ ở xa. Nó có thể mở hoặc truy xuất đến các đối tượng ở xa cũng đơn giản như là truy xuất ngay trên máy tính của mình.

#### 4. Ngôn ngữ lai

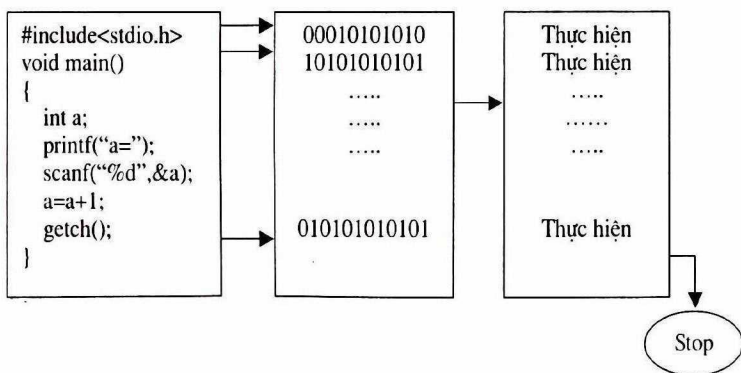
Java là một ngôn ngữ lai vì Java là một ngôn ngữ vừa *biên dịch* và vừa *thông dịch*.

Ta biết rằng hầu hết chương trình máy tính luôn được viết dưới một ngôn ngữ lập trình nào đó, tuy nhiên, máy tính của ta lại không thể thực hiện trực tiếp các chương trình đó vì thực chất máy tính chỉ là sự kết hợp các mạch điện tử và nó chỉ hiểu được *ngôn ngữ máy* (*machine language*). *Ngôn ngữ máy* là ngôn ngữ chỉ được tạo thành từ các số 0 và 1; 0 ứng với mức điện áp thấp và 1 ứng với mức điện áp cao.

Vậy, để một chương trình máy tính viết bằng một ngôn ngữ lập trình nào đó được thực hiện thì trước tiên nó phải được chuyển thành *ngôn ngữ máy*. Quá trình chuyển một chương trình thành *ngôn ngữ máy* đó người ta gọi đó là quá trình *dịch* chương trình.

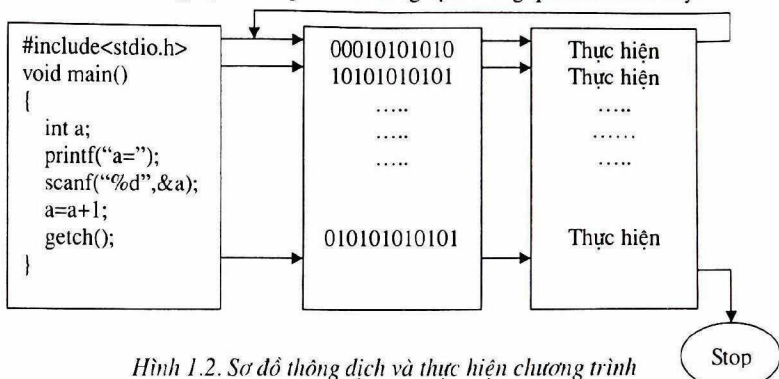
Trong *dịch* chương trình thì có hai cách dịch là *biên dịch* và *thông dịch*.

*Biên dịch* tức là chuyển toàn bộ chương trình sang *ngôn ngữ máy* rồi sau đó mới thực hiện chương trình. Ta có thể hình dung cụ thể qua sơ đồ dưới đây:



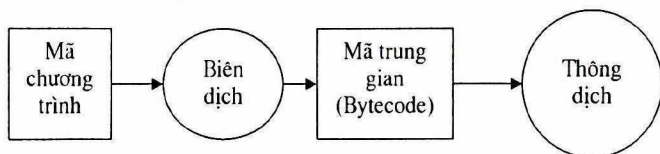
Hình 1.1. Sơ đồ biên dịch và thực hiện chương trình

*Thông dịch* tức là thực hiện dịch từng dòng lệnh của chương trình sau đó sẽ thực hiện ngay dòng lệnh đó rồi lại quay về dịch và thực hiện dòng lệnh tiếp theo. Ta có thể hình dung cụ thể về quá trình thông dịch thông qua hình dưới đây:



Hình 1.2. Sơ đồ thông dịch và thực hiện chương trình

Vậy một chương trình Java vừa được biên dịch vừa được thông dịch như thế nào? Ta hình dung qua sơ đồ dưới đây:



Hình 1.3. Các bước dịch một chương trình Java

Như vậy, để thực hiện một chương trình Java ta cần trải qua hai bước:

Bước 1: *Biên dịch* chương trình Java thành một dạng mã trung gian, được gọi là mã *bytecode*.

Bước 2: *Thông dịch* và thực hiện mã trung gian đó.

## 5. Kiến trúc trung tính

Kiến trúc trung tính của Java nói đến khả năng chạy chương trình Java trên nhiều hệ điều hành và nhiều bộ vi xử lý khác nhau.

Java với khả năng thông dịch mã *bytecode* (được biên dịch từ mã nguồn Java) cho phép tạo ra các máy Java ảo (*JVM - Java Virtual Machine*) trên mỗi hệ thống. Các chương trình Java sẽ chạy trên nền trung tính của các máy ảo Java đó dễ dàng mà không phụ thuộc vào hệ điều hành và bộ vi xử lý.