

KỸ THUẬT LẬP TRÌNH C/C++

Thi-Lan Le

Thi-Lan.Le@mica.edu.vn; lan.lethi1@hust.edu.vn

[Webpage: http://www.mica.edu.vn/perso/Le-Thi-Lan](http://www.mica.edu.vn/perso/Le-Thi-Lan)

Khái niệm về chương trình và lập trình

- ◆ Chương trình: tập các lệnh máy tính phải tuân theo
- ◆ Phần mềm: tập hợp các chương trình
- ◆ Hệ điều hành (HĐH):
 - là một phần mềm
 - cho phép người dùng thao tác với máy tính
 - quản lý tài nguyên (CPU, bộ nhớ, phần cứng/mềm,...)
 - chạy chương trình
- ◆ Thuật toán: chuỗi lệnh có thứ tự để giải quyết một vấn đề nhất định
- ◆ Trình biên dịch: là chương trình cho phép chuyển đổi mã nguồn thành mã máy

Mã lệnh và dữ liệu

- ◆ Máy tính chỉ hiểu mã nhị phân (byte, bit)
 - Một mã lệnh có thể có mã 01000001
 - Chữ cái 'A' có mã 01000001
 - Số 65 có mã 01000001
- ◆ Làm thế nào máy tính hiểu 01000001 biểu diễn gì:
 - Phụ thuộc vào mã lệnh đang chạy là gì
 - Người viết chương trình phải hiểu vùng nhớ đang truy xuất chứa gì
- ◆ Bộ nhớ trong của máy tính chứa cả dữ liệu và mã lệnh của chương trình khi chạy

Ngôn ngữ lập trình

- ◆ Là ngôn ngữ được con thiết kế dùng để tạo ra những chương trình chạy trên máy tính theo ý đồ mong muốn
- ◆ Quá trình phát triển:
 - Mã máy: dùng trực tiếp mã nhị phân, không cần biên dịch, phụ thuộc và bộ vi xử lý
 - Thế hệ thứ 2 (hợp ngữ): cần biên dịch, có thể đọc hiểu được, phụ thuộc và bộ vi xử lý
 - Thế hệ thứ 3 (cấu trúc): cấu trúc điều khiển, kiểu dữ liệu, đóng gói. VD: Fortran, C, C++, Basic, Pascal, COBOL,...
 - Thế hệ thứ 4: nâng cao hiệu quả nhưng giảm các yếu tố dễ gây lỗi, cú pháp gần gũi hơn với ngôn ngữ nói. VD: SQL, LabVIEW, ColdFusion,...

Quá trình phát triển phần mềm

- ◆ Vòng đời của phần mềm:
 - Phân tích yêu cầu của bài toán (problem definition)
 - Thiết kế phần mềm (design)
 - Cài đặt thuật toán (coding)
 - Bảo trì, cập nhật và phát triển tiếp
 - Lỗi thời
- ◆ Thử nghiệm: là quá trình kiểm tra sự hoạt động các tính năng của phần mềm
- ◆ Gỡ lỗi: là quá trình tìm ra nguyên nhân của lỗi, và sửa nó

Lỗi chương trình

◆ Lỗi cú pháp (lỗi biên dịch):

- Do viết chương trình không tuân theo cú pháp quy định
- Được phát hiện bởi trình biên dịch
- Chú ý: đôi khi lỗi không được phát hiện vì bị hiểu sai sang cú pháp khác

◆ Lỗi khi chạy:

- Khi chương trình chạy vi phạm những điều kiện cho phép
- Được phát hiện khi chạy

◆ Lỗi logic:

- Do thuật toán sai
- Máy tính không phát hiện

Lịch sử ngôn ngữ C

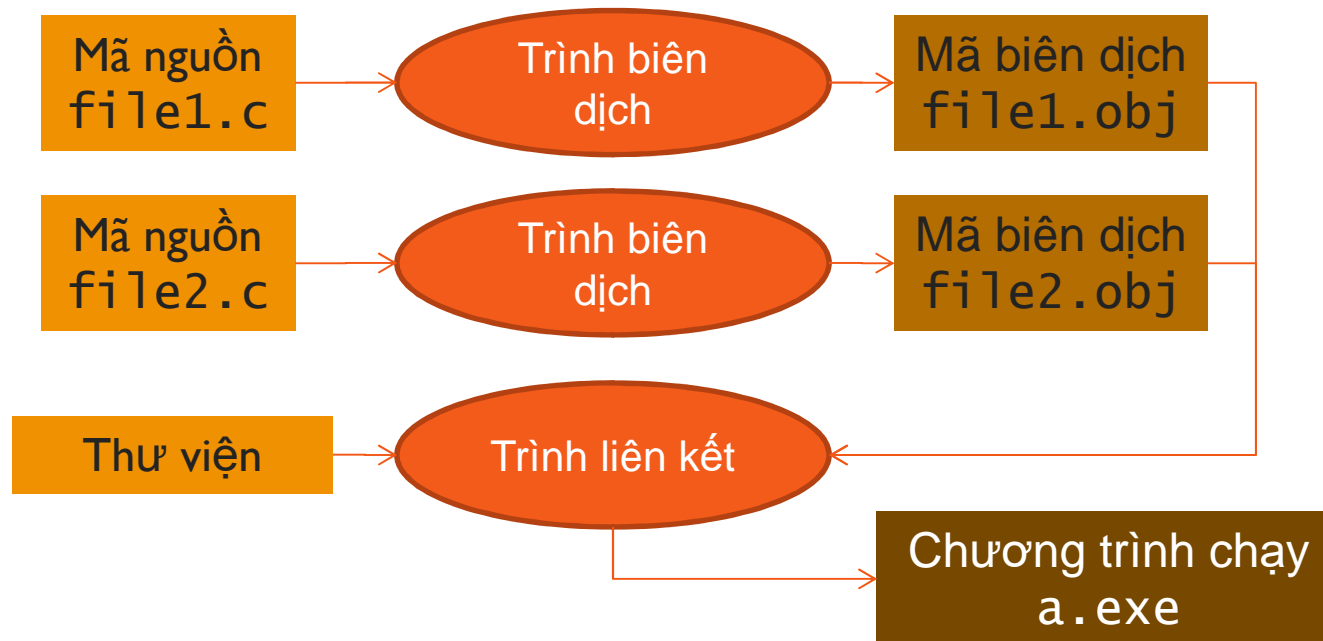
- ◆ Ra đời trong những năm 1970, gắn liền với sự phát triển của HĐH Unix. Tác giả: Dennis Ritchie
 - B là ngôn ngữ lập trình được phát triển tại Bell Labs vào khoảng năm 1969, Nhà phát triển: Ken Thompson, Dennis Ritchie
- ◆ Mục tiêu:
 - Đề cao tính hiệu quả
 - Có khả năng truy xuất phần cứng ở cấp thấp
 - Ngôn ngữ có cấu trúc (thay cho lập trình bằng hợp ngữ)
- ◆ C là ngôn ngữ trung gian giữa cấp thấp...
 - Có khả năng truy xuất bộ nhớ trực tiếp
 - Cú pháp ngắn gọn, ít từ khoá
- ◆ ... và cấp cao
 - Không phụ thuộc phần cứng
 - Cấu trúc, hàm, khả năng đóng gói
 - Kiểm tra kiểu dữ liệu

Lịch sử ngôn ngữ C++

- ◆ Ra đời năm 1979 bằng việc mở rộng ngôn ngữ C. Tác giả: Bjarne Stroustrup
- ◆ Mục tiêu:
 - Thêm các tính năng mới
 - Khắc phục một số nhược điểm của C
- ◆ Bổ sung những tính năng mới so với C:
 - Lập trình hướng đối tượng (OOP)
 - Lập trình tổng quát (template)
 - Nhiều tính năng nhỏ giúp lập trình linh hoạt hơn nữa (thêm kiểu bool, khai báo biến bất kỳ ở đâu, kiểu mạnh, định nghĩa chồng hàm, namespace, xử lý ngoại lệ,...)

Biên dịch chương trình C/C++

- ◆ Là quá trình chuyển đổi từ mã nguồn (do người viết) thành chương trình ở dạng mã máy để có thể thực thi được



Biên dịch chương trình C/C++ (tiếp)

- ◆ Cho phép dịch từng file riêng rẽ giúp:
 - Dễ phân chia và quản lý từng phần của chương trình
 - Khi cần thay đổi, chỉ cần sửa đổi file liên quan
 - giảm thời gian bảo trì, sửa đổi
 - Chỉ cần dịch lại những file có thay đổi khi cần thiết
 - giảm thời gian dịch
- ◆ Các trình biên dịch hiện đại còn cho phép tối ưu hoá dữ liệu và mã lệnh
- ◆ Một số trình biên dịch thông dụng: MS Visual C++, gcc, Intel C++ Compiler, Watcom C/C++,...

Tại sao chọn C/C++

◆ Ưu điểm:

- Hiệu quả
- Linh hoạt, khả năng tùy biến cao
- Được hỗ trợ rộng rãi
 - ◆ trên các môi trường khác nhau
 - ◆ nhiều thư viện và công cụ sẵn có

◆ Nhược điểm:

- Ngôn ngữ [quá] phức tạp
- Khó kiểm soát lỗi hơn so với các ngôn ngữ bậc cao (Java, .NET, script,...), nhất là nguyên nhân từ sử dụng con trỏ

Một vài điểm chú ý về ngôn ngữ C/C++

- ◆ Cú pháp có phân biệt chữ hoa/thường: int, Int, INT là hoàn toàn khác nhau
- ◆ Dấu ; dùng để phân tách các câu lệnh đơn
- ◆ Dấu { ... } để quy định một khối câu lệnh
- ◆ Không được đặt tên biến/hằng/hàm... trùng với từ khoá có sẵn (void, int, char, struct, const,...)
- ◆ Trong một khối lệnh không có cấu trúc định hướng (if, for, while,...) thì các lệnh sẽ thực hiện tuần tự từ trên xuống
- ◆ Chú thích:
 - trong C bằng: /* ... */
 - trong C++ có thêm ký hiệu // để chú thích đến hết dòng

Giới thiệu MS Visual C++

- ◆ Là một môi trường tích hợp (IDE), trợ giúp:
 - Viết chương trình
 - Biên dịch
 - Gỡ lỗi (debug)

- ◆ Làm ví dụ...