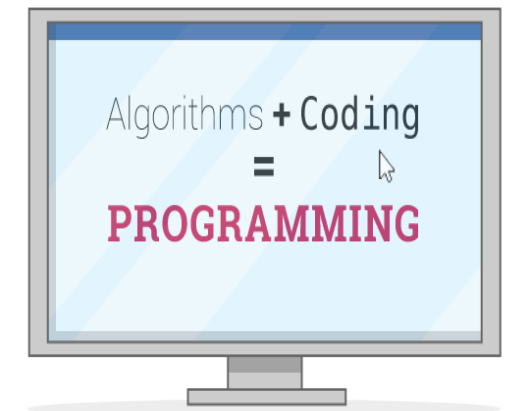
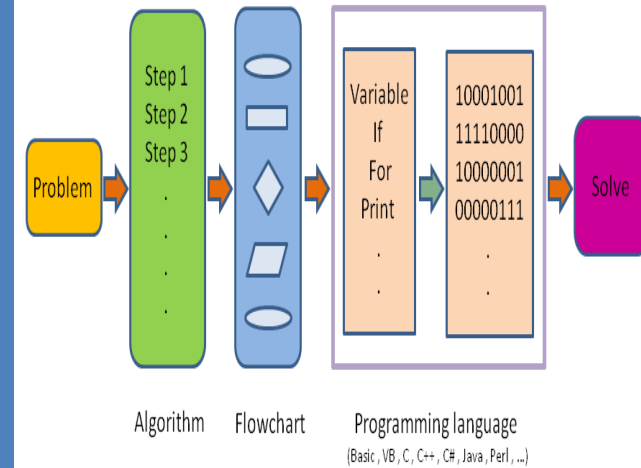


NHẬP MÔN LẬP TRÌNH

Chương 1 - TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

GV: Lê Thị Bích Hằng

Trường Đại học Nha Trang



TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình

Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Lập trình máy tính (Computer programming)

- Gọi tắt là lập trình (programming).
- Kỹ thuật **cài đặt** một hoặc nhiều **thuật toán** trừu tượng có liên quan với nhau bằng một **ngôn ngữ lập trình** để tạo ra một **chương trình máy tính**.

■ Thuật toán/Giải thuật (Algorithm)

- Là **tập hợp (dãy) hữu hạn** các **chỉ thị** (hành động) được **định nghĩa rõ ràng** nhằm giải quyết một bài toán cụ thể nào đó.

■ Chương trình máy tính (Computer program)

- Là một tập hợp những **câu lệnh** được viết bằng một hoặc nhiều ngôn ngữ lập trình **theo một trật tự xác định** nhằm **tự động thực hiện** một số chức năng hoặc giải quyết một vấn đề cụ thể nào đó.
- Phần mềm thực hiện các chức năng của nó bằng cách gửi các chỉ thị trực tiếp đến phần cứng, hoặc bằng cách cung cấp dữ liệu để phục vụ các chương trình hay phần mềm khác.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình

Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình

Tổng quan về ngôn
ngữ C



TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Thuật toán/Giải thuật (Algorithm)

- Là tập hợp (dãy) hữu hạn các chỉ thị (hành động) được định nghĩa rõ ràng nhằm giải quyết một bài toán cụ thể nào đó.

■ Các tính chất của giải thuật

- **Tính chính xác:** quá trình tính toán hay các thao tác máy tính thực hiện là chính xác.
- **Tính rõ ràng:** các câu lệnh minh bạch được sắp xếp theo thứ tự nhất định.
- **Tính khách quan:** được viết bởi nhiều người trên máy tính nhưng kết quả phải như nhau.
- **Tính phổ dụng:** có thể áp dụng cho một lớp các bài toán có đầu vào tương tự nhau.
- **Tính kết thúc:** giải thuật phải dừng lại sau một số bước tính toán.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Biểu diễn giải thuật

- **Ngôn ngữ tự nhiên** (Natural language): sử dụng ngôn ngữ thường ngày để liệt kê các bước của giải thuật.
- **Lưu đồ** (Flowchart): sử dụng các sơ đồ hình khối để mô tả giải thuật, mỗi hình khối quy định một hành động.
- **Mã giả** (Pseudo code): sử dụng ngôn ngữ tự nhiên kết hợp với ngôn ngữ lập trình.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Biểu diễn giải thuật bằng ngôn ngữ tự nhiên

- Ví dụ: Giải thuật giải phương trình bậc nhất $ax + b = 0$

Đầu vào: a, b thuộc \mathbb{R}

Đầu ra: nghiệm phương trình $ax + b = 0$

1. Nhập 2 số thực a và b
2. Nếu $a = 0$ thì
 - 2.1. Nếu $b = 0$ thì
 - 2.1.1. Phương trình vô số nghiệm
 - 2.1.2. Kết thúc thuật toán.
 - 2.2. Ngược lại
 - 2.2.1. Phương trình vô nghiệm.
 - 2.2.2. Kết thúc thuật toán.
3. Nếu $a \neq 0$ thì
 - 3.1. Phương trình có nghiệm
 - 3.2. Giá trị của nghiệm đó là $x = -b/a$
 - 3.3. Kết thúc thuật toán.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

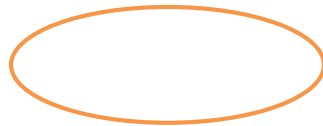
Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Biểu diễn giải thuật bằng lưu đồ

- Một số hình khối phổ biến:



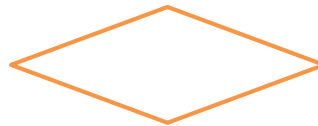
Khối giới hạn

Chỉ thị bắt đầu và kết thúc



Khối vào ra

Nhập/Xuất dữ liệu



Khối lựa chọn

Tùy điều kiện sẽ rẽ nhánh



Khối thao tác

Ghi thao tác cần thực hiện



Đường đi

Chỉ hướng thao tác tiếp theo

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

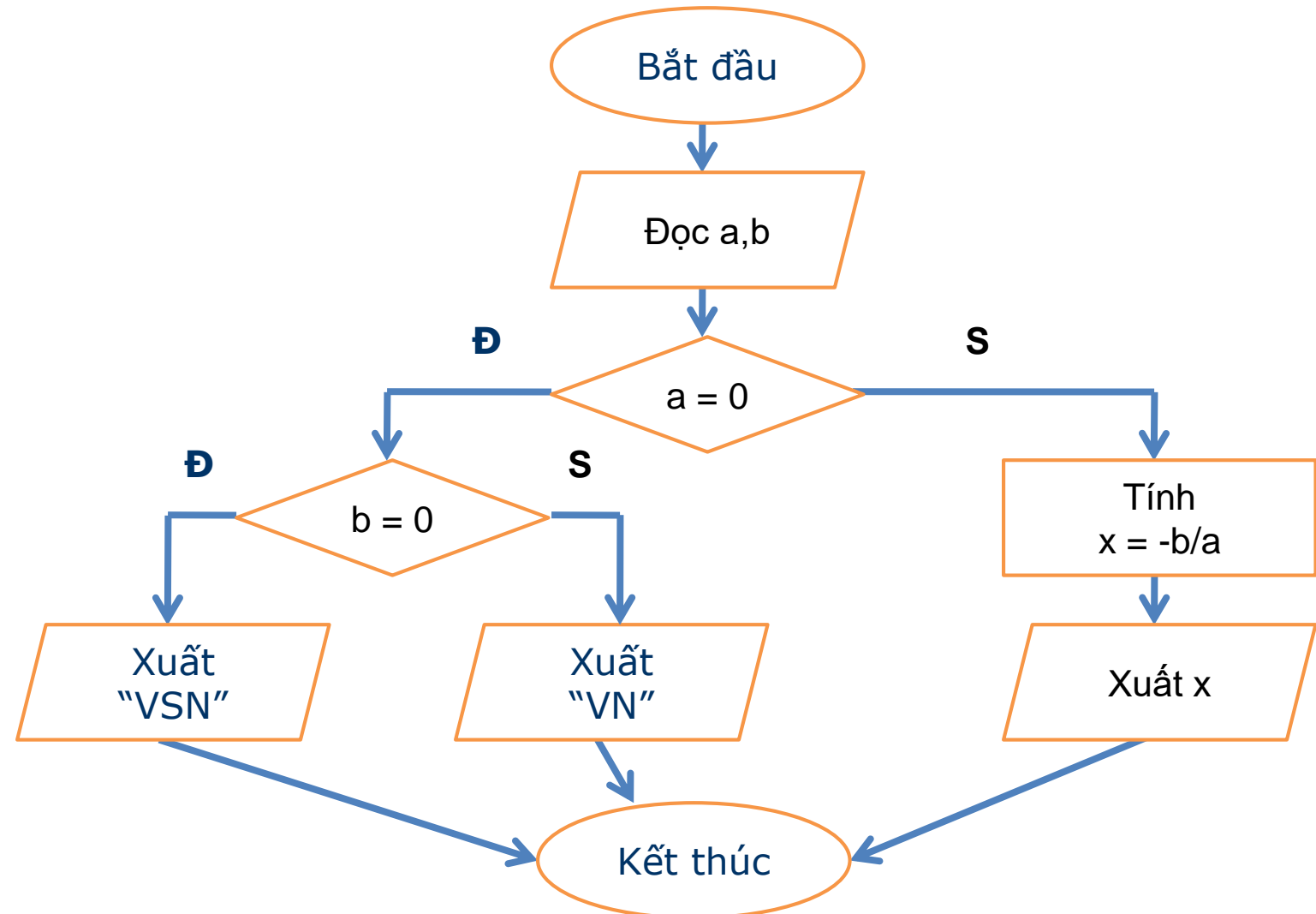
Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Biểu diễn giải thuật bằng lưu đồ

- Ví dụ: Giải thuật giải phương trình bậc nhất $ax + b = 0$



TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Biểu diễn giải thuật bằng mã giả

- Ví dụ: Giải thuật giải phương trình bậc nhất $ax + b = 0$

Đầu vào: a, b thuộc \mathbb{R}

Đầu ra: nghiệm phương trình $ax + b = 0$

```
if a = 0 then
    if b = 0 then
        print "Phương trình vô số nghiệm"
    else
        print "Phương trình vô nghiệm"
    endif
else
    begin
        x := -b/a
        print "Phương trình có nghiệm là x"
    end
endif
```

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Giới thiệu


- Giải thuật phải được thể hiện dưới dạng **các dòng lệnh theo các quy ước nào đó mà máy tính có thể thực hiện được** một cách trực tiếp. Tập các kí hiệu và các quy tắc viết các lệnh để thể hiện giải thuật được gọi là ngôn ngữ lập trình (*programming language*).
- Ngôn ngữ lập trình được dùng để tạo ra các chương trình máy tính nhằm mục đích điều khiển máy tính hoặc mô tả các giải thuật để người khác có thể đọc hiểu.
- Mỗi ngôn ngữ lập trình có **một bộ từ khóa** duy nhất (*keyword*) và **một cú pháp** riêng (*syntax*) để tổ chức các câu lệnh của chương trình.
- Mỗi chương trình sẽ mang một ý nghĩa nhất định được gọi là **ngữ nghĩa** (*semantic*) của chương trình.

■ Phân loại ngôn ngữ lập trình (NNLT):

- Ngôn ngữ máy (Machine Language)
- Hợp ngữ (Assembly Language)
- Ngôn ngữ cấp cao (High level language)

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Phân loại NNLT: Ngôn ngữ máy

- Là các chỉ thị dưới dạng nhị phân, can thiệp trực tiếp vào trong các mạch điện tử.
- Ngôn ngữ máy là ngôn ngữ duy nhất mà bộ vi xử lý (CPU) có thể nhận biết, hiểu được và thực hiện trực tiếp mà **không cần dịch** sang bất kỳ một ngôn ngữ nào khác => Chương trình có thể được thực hiện ngay không cần qua bước trung gian nào.
- Chương trình viết bằng ngôn ngữ máy dễ sai sót, cồng kềnh và khó đọc, khó hiểu.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ bản về lập trình
Quy trình xây dựng một chương trình máy tính
Giải thuật, biểu diễn giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn ngữ C




■ Phân loại NNLT: Hợp ngữ

- Có các cấu trúc rất giống với ngôn ngữ máy, tuy nhiên có thể viết lệnh dưới dạng mã chữ.
- Mã chữ thể hiện mã lệnh hoặc các đối tượng trong lệnh.
- Để một chương trình viết bằng hợp ngữ chạy được trên máy tính, nó cần phải được **dịch ra ngôn ngữ máy**.
- Ví dụ minh họa phép tính cộng hai số:

Location Hex	Machine language	Assembly language	Comments
100	0010 0001 0000 0100	LDA 104	Load first operand into AC
101	0001 0001 0000 0101	ADD 105	Add second operand to AC
102	0011 0001 0000 0110	STA 106	Store sum in location 106
103	0111 0000 0000 0001	HLT	Halt computer
104	0000 0000 0101 0011	Operand	83 decimal
105	1111 1111 1111 1110	Operand	-2 decimal
106	0000 0000 0000 0000	Operand	Store sum here

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C




■ Phân loại NNLT: Ngôn ngữ cấp cao

- Gần với ngôn ngữ con người (Anh ngữ).
- Được phát triển rất đa dạng và phong phú theo hướng: lập trình cấu trúc, lập trình hướng đối tượng, lập trình logic...
- Một chương trình viết bằng ngôn ngữ cấp cao được gọi là **chương trình nguồn** (source program).
- Để máy tính thực hiện được các lệnh trong chương trình nguồn thì phải có một **chương trình dịch** để dịch chương trình nguồn thành dạng chương trình có khả năng thực thi.
- Ví dụ minh họa phép tính cộng hai số bằng ngôn ngữ C:

```
1 #include <stdio.h>
2 int main()
3 {   int x, y, z;
4     x = 83;
5     y = -23;
6     z = x + y;
7     printf("\n %d + %d = %d", x, y, z);
8     return 0;
9 }
```

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Các mô hình lập trình

• Lập trình hướng thủ tục

- Một nhiệm vụ lớn được **phân thành nhiều công việc nhỏ** hơn, được chi tiết và cụ thể hoá để được các vấn đề đơn giản hơn.
- Cách thức phân tích và thiết kế như vậy được gọi là **nguyên lý lập trình từ trên xuống** (*top-down*), để thể hiện quá trình suy diễn từ cái chung cho đến cái cụ thể.
- Một số ngôn ngữ lập trình cấu trúc phổ biến: C, Pascal, Foxpro.

• Lập trình logic


- Là một phương pháp tiếp cận việc **biểu diễn tri thức** và giải các bài toán logic từ một cơ sở tri thức cho trước trên máy tính.
- Một cơ sở tri thức là một tập các sự kiện và các luật biểu diễn quan hệ logic giữa các sự kiện đó. Lập trình logic xuất phát từ một cơ sở tri thức và một câu hỏi, tiến hành các lập luận logic để tìm ra lời giải cho câu hỏi đó.
- Một số ngôn ngữ lập trình logic thông dụng: Prolog, ASP, Datalog.

• Lập trình hướng đối tượng

- Lập trình hướng đối tượng là phương pháp lập trình **xây dựng trên mô hình thế giới thực**, khi mà mọi sự vật được xem như các đối tượng. Các đối tượng bao gồm các thuộc tính và các hành động của riêng nó, cho phép tái sử dụng mã nguồn, có tính bảo mật cao và tính nhất quán dữ liệu.
- Một số ngôn ngữ lập trình hướng đối tượng thông dụng: C++, C#, Java, Python.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Chương trình dịch

- Cho dù chương trình được viết dưới dạng ngôn ngữ cấp cao nào, thì cuối cùng **chương trình đó cũng cần được chuyển đổi sang ngôn ngữ máy** để máy tính có thể hiểu được ngôn ngữ đó.
- Chương trình dùng để chuyển đổi mã nguồn thành mã máy được gọi là chương trình dịch (*translator program*).
- Có hai loại chương trình dịch:
 - Trình biên dịch (*Compiler*)
 - Trình thông dịch (*Interpreter*).

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình

Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

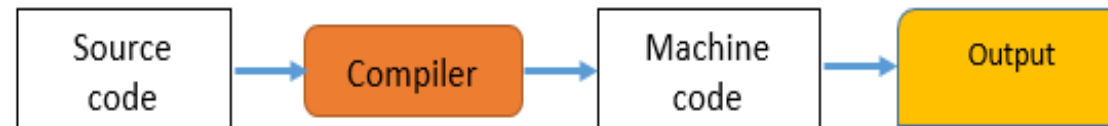
Ngôn ngữ lập trình

Tổng quan về ngôn
ngữ C



■ Chương trình dịch: Trình biên dịch

- Trình biên dịch là một chương trình máy tính, **chuyển đổi toàn bộ mã nguồn thành mã máy trước** khi chương trình thực thi (nghĩa là trình biên dịch sẽ tạo ra một tập tin thực thi).



- Một tập tin thực thi là một tập tin sẵn sàng **thực hiện bất cứ lúc nào** và nó không thể dễ sửa đổi. Vì vậy, mỗi khi lập trình viên thay đổi chương trình thì phải biên dịch lại mã nguồn.
- Một số ngôn ngữ lập trình thông dụng sử dụng trình biên dịch: C, C++, C#, Pascal.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình

Quy trình xây dựng
một chương trình
máy tính

Giải thuật, biểu diễn
giải thuật

Ngôn ngữ lập trình

Tổng quan về ngôn
ngữ C



■ Chương trình dịch: Trình thông dịch

- Trình thông dịch là một chương trình máy tính, **chuyển đổi từng câu lệnh của mã nguồn thành mã máy** trong quá trình chương trình thực thi.



- Với trình thông dịch, **các câu lệnh được thực hiện ngay khi chúng được dịch**, và lỗi có thể được phát hiện nhanh chóng: một khi lỗi được tìm thấy, chương trình sẽ ngừng chạy và người dùng được thông báo tại phần nào của chương trình mà quá trình thông dịch bị thất bại.
- Một số ngôn ngữ lập trình thông dụng sử dụng trình thông dịch: Python, Ruby, PHP, Perl, JavaScript.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C




■ Chương trình dịch: Các điểm khác nhau cơ bản giữa trình biên dịch và trình thông dịch

Interpreter (Trình thông dịch)	Compiler (Trình biên dịch)
Dịch mỗi lần một câu lệnh của chương trình nguồn thành mã máy.	Dịch toàn bộ câu lệnh của chương trình nguồn thành mã máy.
Mất ít thời gian hơn để phân tích mã nguồn nhưng thời gian thực thi chậm hơn.	Mất nhiều thời gian hơn để phân tích mã nguồn nhưng thời gian thực thi tương đối nhanh hơn.
Không có mã đối tượng trung gian được tạo ra, do đó ít tốn bộ nhớ hơn.	Tạo mã đối tượng trung gian để liên kết, do đó cần nhiều bộ nhớ hơn.
Tiếp tục dịch chương trình cho đến khi gặp lỗi đầu tiên, trong trường hợp đó chương trình sẽ dừng lại. Do đó việc gỡ lỗi dễ dàng hơn.	Trình biên dịch chỉ đưa ra thông báo lỗi sau khi xét toàn bộ chương trình. Do đó việc gỡ lỗi tương đối khó hơn.
Không tạo chương trình thực thi.	Tạo chương trình thực thi (dưới dạng exe) có thể chạy độc lập với chương trình gốc.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Giới thiệu

- Ngôn ngữ lập trình C do Dennis Ritchie phát triển tại phòng thí nghiệm Bell (Mỹ) vào năm 1972.
- Tiền thân là ngôn ngữ B, do KenThompson phát triển (cũng tại Bell).
- Là ngôn ngữ lập trình có cấu trúc.

■ Đặc điểm của ngôn ngữ lập trình C

- **Tính cô đọng (compact)**: C chỉ có 32 từ khóa chuẩn và 40 toán tử chuẩn, được biểu diễn bằng những chuỗi ký tự ngắn gọn.
- **Tính cấu trúc (structured)**: C có một tập hợp những chỉ thị của lập trình như cấu trúc lựa chọn, lặp...
- **Tính tương thích (compatible)**: C có bộ tiền xử lý và một thư viện chuẩn vô cùng phong phú nên khi chuyển từ máy tính này sang máy tính khác các chương trình viết bằng C vẫn hoàn toàn tương thích.
- **Tính linh động (flexible)**: C là một ngôn ngữ rất uyển chuyển về cú pháp, chấp nhận nhiều cách thể hiện, có thể thu gọn kích thước của các mã lệnh làm chương trình chạy nhanh hơn.
- **Biên dịch (compile)**: Cho phép biên dịch nhiều tập tin chương trình riêng rẽ thành các tập tin đối tượng (object) và liên kết (link) các đối tượng đó lại với nhau thành một chương trình có thể thực thi được (executable) thống nhất.

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C



■ Cấu trúc cơ bản của một chương trình C

- Một chương trình C về cơ bản bao gồm các thành phần sau:
 - Các chỉ thị tiền xử lý (*preprocessor*)
 - Các hàm (*function*)
 - Các biến (*variable*)
 - Các câu lệnh (*statement*)
 - Các biểu thức (*expression*)
 - Các lời chú thích (*comment*).

• Ví dụ:

	1	/* Chương trình tính tổng hai số nguyên*/	Comment
Preprocessor	2	#include <stdio.h>	
	3	int main()	
Function	4	{	
	5	int x, y, z;	
Variables	6	x = 83;	
	7	y = -23;	
Statement	8	z = x + y;	
	9	printf("\n %d + %d = %d", x, y, z);	Statement
	10	return 0;	
	11	}	

TỔNG QUAN VỀ LẬP TRÌNH MÁY TÍNH

Các khái niệm cơ
bản về lập trình
Quy trình xây dựng
một chương trình
máy tính
Giải thuật, biểu diễn
giải thuật
Ngôn ngữ lập trình
Tổng quan về ngôn
ngữ C

■ Quy trình biên dịch một tập tin nguồn thành tập tin thực thi trong C

- ✓ **B1. Trình biên dịch (Compiler)** kiểm tra mã nguồn. Nếu chương trình nguồn có lỗi cú pháp (*syntax error*), trình biên dịch sẽ đưa ra các thông báo lỗi, lập trình viên phải sửa các lỗi đó cho đến khi chương trình nguồn không còn lỗi nữa.
- ✓ **B2. Trình biên dịch tạo tập tin đối tượng** có phần mở rộng là *obj* (trong hệ điều hành Windows) chứa các dãy lệnh ở dạng mã máy.
- ✓ **B3. Trình liên kết (Linker)** liên kết tập tin đối tượng với các tập tin thư viện tạo thành tập tin thi hành có phần mở rộng là *exe* (trong Windows).

