



TIN HỌC CƠ SỞ

Bài 3. Phần mềm

NỘI DUNG

- Ngôn ngữ lập trình
- Ngôn ngữ máy
- Hợp ngữ
- Ngôn ngữ thuật toán
- Quá trình thực hiện một chương trình trên ngôn ngữ bậc cao
- Môi trường phát triển tích hợp

Mô đun 9. NGÔN NGỮ LẬP TRÌNH VÀ CHƯƠNG TRÌNH DỊCH

Giảng viên: TS. TRẦN THỊ MINH CHÂU
Email: chauttm@vnu.edu.vn



NGÔN NGỮ LẬP TRÌNH

- Ngôn ngữ: phương tiện trao đổi thông tin
- Ngôn ngữ lập trình (programming language) là ngôn ngữ dùng trao đổi thuật toán hướng dẫn máy tính thực hiện các công việc đã định.
- Hai khía cạnh của ngôn ngữ:
 - Cú pháp (syntax): các quy tắc viết đúng
 - Ngữ nghĩa (semantic): ý nghĩa mà ngôn ngữ chuyển tải
- Một chương trình máy tính (program) phải được thể hiện trên một ngôn ngữ xác định.
- Một thuật toán có thể diễn đạt bằng nhiều chương trình khác nhau trên những ngôn ngữ khác nhau.



CÁC MỨC CỦA NGÔN NGỮ LẬP TRÌNH

- Ngôn ngữ máy: ngôn ngữ thể hiện trực tiếp trong hệ lệnh của máy. Nói chung ngôn ngữ máy là ngôn ngữ ở mức các bit, nên cũng được gọi là ngôn ngữ nhị phân
- Hợp ngữ (assembly) là loại ngôn ngữ về cơ bản là gần với ngôn ngữ nhị phân, mỗi lệnh của ngôn ngữ máy có một lệnh tương ứng của hợp ngữ nhưng hợp ngữ sử dụng mã chữ
- Ngôn ngữ bậc cao – còn gọi là ngôn ngữ thuật toán (Algorithmic language) là ngôn ngữ biểu diễn thuật toán độc lập với hệ lệnh của máy được viết dưới dạng gần với ngôn ngữ tự nhiên và ngôn ngữ toán học



NGÔN NGỮ MÁY

- Chính là ngôn ngữ được viết bằng lệnh máy trong hệ nhị phân hoặc hệ 16
- Ưu điểm, tận dụng được khả năng của máy, tối ưu được thời gian chạy
- Nhược điểm: khó viết, khó chữa lỗi, phụ thuộc vào từng loại máy. Nói chung chi phí phát triển phần mềm trên ngôn ngữ máy là cao.

Mã máy nhị phân	Mã hexa	Ý nghĩa
1001 0001 0110 0000 0001 0000	A1 60 10	Nạp 1060 lên TG AX
0000 0011 0110 0110 0001 0000	03 66 10	Cộng AX với 1066 -> AX
1010 0011 0000 0000 0010 1011	A3 00 2B	Ghi từ AX về 2B00



HỢP NGỮ (ASSEMBLY)

- Về cơ bản, mỗi lệnh hợp ngữ tương tự với một lệnh máy – nhưng dùng mã chữ nên dễ hiểu, dễ sửa.
- Có các lệnh macro, tương đương với một đoạn chương trình nhiều lệnh máy.
- Ưu điểm: dễ lập trình dễ sửa lỗi hơn ngôn ngữ máy
- Nhược điểm: vẫn còn phức tạp và phụ thuộc vào máy

Hợp ngữ	Mã máy trong hệ hexa
MOV AX CHIEU_DAI	A1 64 10
ADD AX CHIEU_RONG	03 66 10
MOV NUA_CHU_VI AX	A3 00 2B



BỘ DỊCH HỢP NGỮ (ASSEMBLER)

- Máy tính không thể chạy trực tiếp trên mã hợp ngữ.
- Phải dịch chương trình trên hợp ngữ thành một chương trình trên ngôn ngữ máy nhờ một phần mềm có tên là bộ dịch hợp ngữ hay hợp dịch (assembler)
- Assembler phải bố trí không gian nhớ cho các đối tượng, sau đó thay thế mã lệnh và địa chỉ bằng các mã số. Việc thay thế cũng được thực hiện với các lệnh macro, mỗi macro tương đương với nhiều lệnh máy.
- Kết quả của bước dịch đầu tiên là tạo ra các mô đun đối tượng - là các đoạn chương trình dưới dạng nhị phân.
- Các đoạn chương trình ở mã nhị phân có thể được ghép với nhau, để tạo thành chương trình hoàn chỉnh



NGÔN NGỮ THUẬT TOÁN

- Ngôn ngữ máy và hợp ngữ phụ thuộc vào máy, lại khó dùng, vì nó buộc người lập trình phải viết tinh tế đến mức lệnh máy.
- Ngôn ngữ thuật toán (algorithmic language) chỉ diễn tả thuật toán mà thôi, không phụ thuộc vào hệ lệnh đặc thù của máy tính cụ thể.
- Ngôn ngữ thuật toán có hình thức giống với ngôn ngữ tự nhiên hoặc ngôn ngữ toán học nên dễ diễn đạt hơn nhiều so với ngôn ngữ máy hoặc hợp ngữ



VÍ DỤ VỀ NGÔN NGỮ BẬC CAO

- Ví dụ giải phương trình bậc 2 trên PASCAL

```
DELTA := B*B - 4*A*C;  
IF DELTA >= 0 THEN  
  BEGIN  
    X1 := (- B + SQRT(DELTA))/(2*A);  
    X2 := (- B - SQRT(DELTA))/(2*A);  
  
    WRITE (X1,X2);  
  END  
ELSE  
  WRITE('VO NGHIEM');
```

- Và trên FORTRAN

```
DELTA = B*B - 4* A*C  
IF DELTA < 0 GOTO 10  
X1= (- B + SQRT(DELTA))/(2*A)  
X2 =(- B - SQRT(DELTA))/(2*A)  
WRITE (3,20) X1, X2  
20  FORMAT ('NGHIEM 1= ', F8.3,  
          NGHIEM 2 = ', F8.3)  
GOTO 30  
10  WRITE(3,40)  
40  FORMAT('VO NGHIEM')  
30  STOP  
END
```



THỰC HIỆN CHƯƠNG TRÌNH TRÊN NGÔN NGỮ THUẬT TOÁN

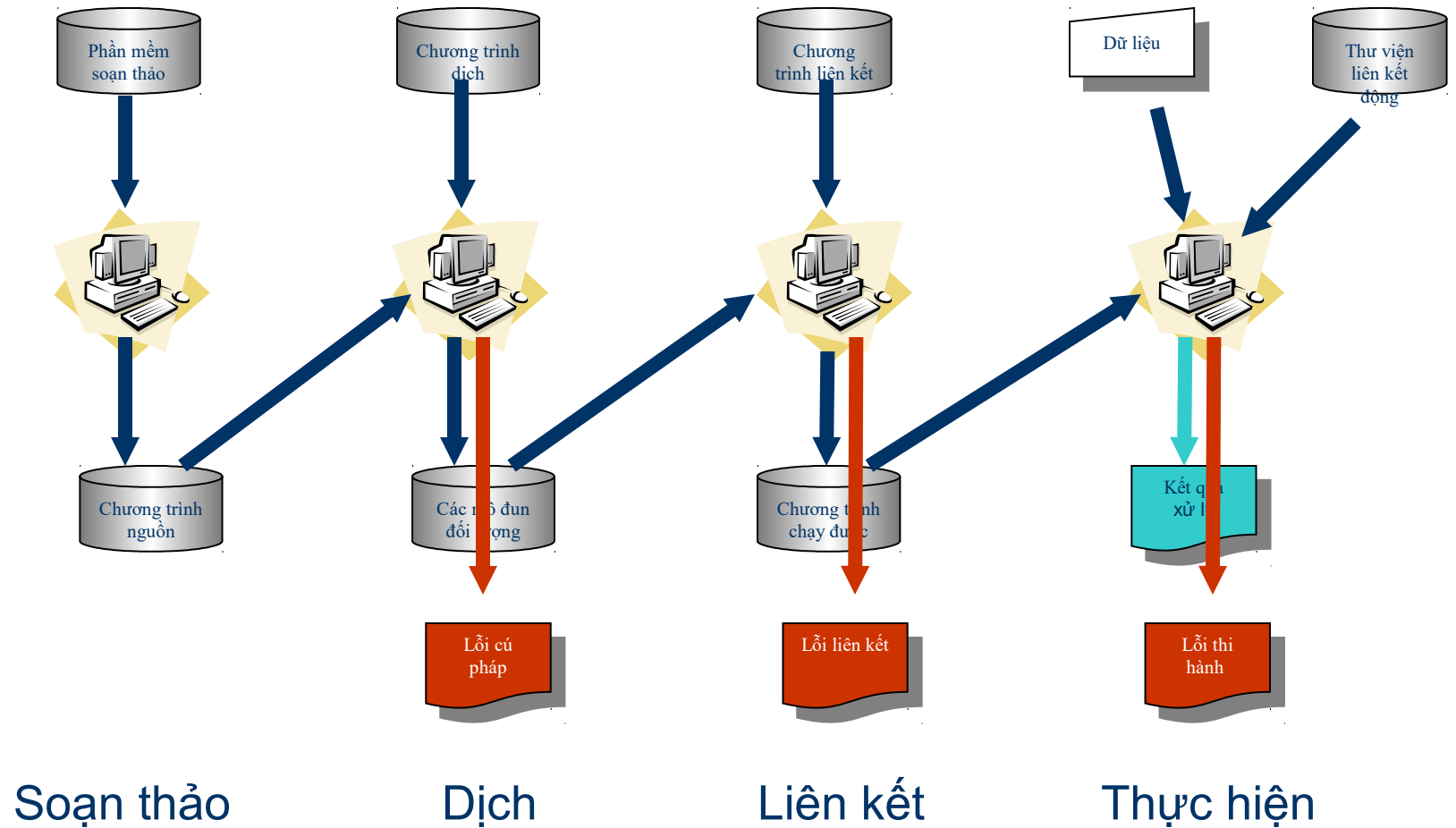
- Máy tính chỉ có thể thi hành trực tiếp ngôn ngữ nhị phân, do đó phải dịch bằng một cách nào đó để máy tính có thể thực hiện được.
- Hai cách thực hiện:
 - Dịch toàn bộ chương trình ra mã nhị phân rồi thực hiện: chương trình này gọi là bộ **biên dịch** (compiler).
 - Không dịch ra mã máy mà dùng một chương trình phân tích và thi hành luôn các lệnh của chương trình nguồn. Chương trình này gọi là bộ **thông dịch** (interpreter).
 - Chương trình thông dịch đóng vai trò của một máy ảo.



THỰC HIỆN CHƯƠNG TRÌNH TRONG CHẾ ĐỘ BIÊN DỊCH

- Soạn thảo chương trình nhờ một bộ soạn thảo nào đó
- Phân tích cú pháp (syntax analys) đảm bảo rõ nghĩa, không nhập nhằng. Nếu không đúng cú pháp sẽ không thể tạo được mã. Tất cả các lỗi cú pháp đều được phát hiện được trong khi dịch.
- Dịch (compile): Tạo ra các đoạn chương trình trên ngôn ngữ máy (gọi là các mô đun đối tượng)
- Liên kết: (link) kết nối các mô đun đối tượng thành một chương trình hoàn chỉnh và duy nhất.
- Tải chương trình vào máy để thực hiện, và nạp dữ liệu để chạy. Khi chạy vẫn còn có thể có lỗi ngữ nghĩa.
- Trong công nghệ hướng tới thành phần người ta có thể thực hiện quá trình liên kết ngay trong quá trình chạy. Các module đã dịch dưới dạng nhị phân được tải vào máy theo lệnh gọi – phải phân phối động bộ nhớ ở mức hệ điều hành
- Gỡ lỗi. Khi gặp lỗi, phân tích để định vị điểm gây lỗi

THỰC HIỆN CHƯƠNG TRÌNH TRÊN NGÔN NGỮ BẬC CAO (tiếp)





MÔI TRƯỜNG TÍCH HỢP

- Trước đây các quy trình soạn thảo, dịch, liên kết, thi hành và gỡ lỗi là những công việc rời rạc. Điều này làm cho việc phát triển phần mềm kém hiệu quả.
- Vào 1985 Turbo Pascal đã khởi đầu một khuynh hướng mới về việc tạo ra các môi trường phát triển tích hợp IDE (Intergated Development Environment): toàn bộ các quá trình soạn thảo, dịch, liên kết, thi hành và gỡ lỗi được thực hiện trong cùng một môi trường
- Một bước phát triển tiếp của IDE là việc phát triển hướng đối tượng, phát triển theo mẫu, làm việc sinh mã chương trình trở nên hiệu quả hơn rất nhiều. Lập trình hướng đối tượng giúp kiểm soát tốt hơn
- Các hệ CASE (Computer Aided Software Engineering) còn cho phép phát sinh mã trên nền thiết kế là một bước tiến theo một khuynh hướng tích hợp việc tạo mã từ quá trình thiết kế.



HẾT BÀI 9. NGÔN NGỮ LẬP TRÌNH

CẢM ƠN ĐÃ THEO DÕI BÀI GIẢNG