



BÀI GIẢNG TIN HỌC CƠ SỞ



BÀI 9. NGÔN NGỮ LẬP TRÌNH VÀ CHƯƠNG TRÌNH DỊCH



Giảng viên: ĐÀO KIẾN QUỐC
Mobile 098.91.93.980
Email: dkquoc@vnu.edu.vn

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

NGÔN NGỮ LẬP TRÌNH

- Ngôn ngữ lập trình (programming language) là ngôn ngữ biểu diễn thuật toán dùng để điều khiển máy tính thực hiện các công việc đã định.
- Các quy tắc viết được gọi là cú pháp (syntax) của ngôn ngữ. ý nghĩa mà ngôn ngữ chuyển tải gọi là ngữ nghĩa (semantic)
- Một chương trình máy tính (program) phải được thể hiện trên một ngôn ngữ xác định. Như vậy 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
- Mỗi ngôn ngữ xác định một kiểu diễn đạt kịch bản điều khiển máy tính. Mỗi một kịch bản điều khiển máy viết trên một ngôn ngữ lập trình gọi là một chương trình (program)

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 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.
- Phải dịch ra ngôn ngữ máy (thay mã lệnh và địa chỉ)
- Có các lệnh macro, cho phép thay thế hiệu quả hơn
- 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

DỊCH HỢP NGỮ (ASSEMBLY)

- Máy không thể chạy trực tiếp được trên mã hợp ngữ. Để máy có thể chạy được thì phải dịch chương trình trên hợp ngữ thành một chương trình trên ngôn ngữ máy. Việc này được thực hiện một cách tự động nhờ một phần mềm có tên là bộ hợp dịch (assembler)
- Trong bước dịch đầu tiên bộ hợp dịch sẽ 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, là các lệnh tương đương với nhiều lệnh. Kết quả của bước dịch đầu tiên là tạo ra các mô đun đối tượng, là các đoạn chương trình dưới dạng nhị phân nhưng chưa có cấu trúc hoàn chỉnh để sẵn sàng chạy ngay.
- Thường sẽ phải thực hiện một bước khác là liên kết, để kết hợp nhiều mô đun đối tượng thành một chương trình nhị phân hoàn chỉnh. Sau đó mới nạp chương trình này vào thi hành

NGÔN NGỮ BẬC CAO

- 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.
- Người ta muốn các ngôn ngữ chỉ diễn tả thuật toán mà thôi, không liên quan đến các hệ lệnh đặc thù của máy tính cụ thể. Các ngôn ngữ này gọi là ngôn ngữ bậc cao (high level language) hay còn gọi là ngôn ngữ thuật toán (algorithmic language)
- 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('Vô nghiệm')
```

- 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  END
```

DỊCH NGÔN NGỮ BẬC CAO

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.

Có hai cách thực hiện:

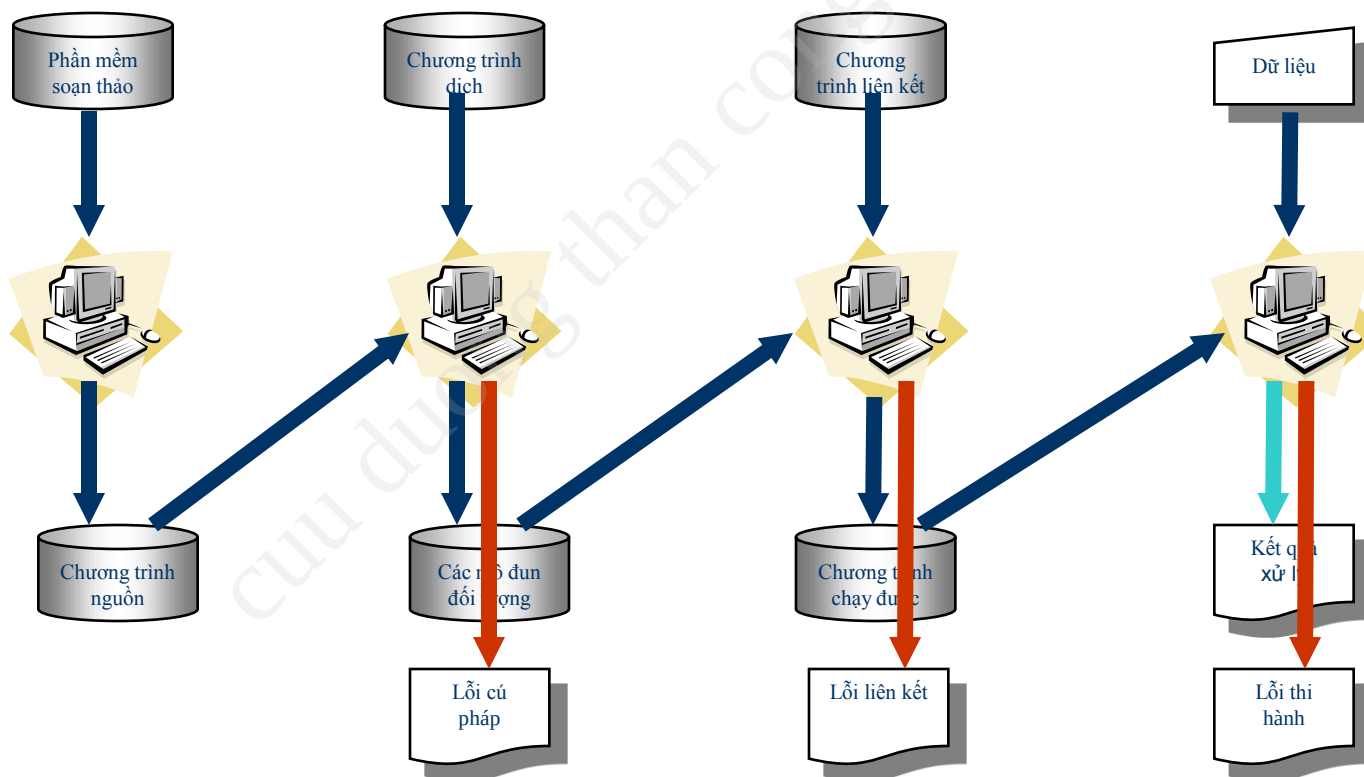
- Sử dụng một chương trình mô phỏng (phần mềm này đã ở mã nhị phân gọi là **chương trình thông dịch** - interpreter). Chương trình này đọc và thi hành các lệnh trong ngôn ngữ bậc cao. Do vậy chương trình thông dịch thực sự đóng vai trò một máy ảo. Trong chế độ thông dịch, không sinh chương trình tương ứng trong mã nhị phân
- Dịch chương trình trong ngôn ngữ thuật toán thành một chương trình ở ngôn ngữ máy bảo toàn ngữ nghĩa nhờ **chương trình biên dịch** (compiler)

THỰC HIỆN CHƯƠNG TRÌNH TRÊN NGÔN NGỮ BẬC CAO



- Soạn thảo chương trình nhờ một bộ soạn thảo nào đó
- Phân tích từ vựng (lexical analysis): tạo ra hồ sơ của tất cả các đối tượng của chương trình phục vụ cho việc phân phối không gian nhớ sau này
- Phân tích cú pháp (syntax analysis): Cú pháp (syntax): quy tắc viết các câu lệnh (statement) đảm bảo rõ nghĩa, không nhập nhằng. Nếu không đúng sẽ không thể tạo được mã Tất cả các lỗi cú pháp đều được phát hiện được trong khi dịch.
- Tạo mã, tối ưu mã (code generation, optimization)
- Liên kết: (link) kết nối các mô đun đối tượng thành một chương trình hoàn chỉnh và duy nhất.
- Thực hiện, tải chương trình và nạp dữ liệu để chạy. Khi chạy vẫn còn có thể có lỗi ngữ nghĩa. Lỗi ngữ nghĩa chỉ có thể phát hiện khi chạy chương trình

DỊCH SANG NGÔN NGỮ MÁY



Soạn thảo

Dịch

Liên kết

Thực hiện

MÔI TRƯỜNG PHÁT TRIỂN PHẦN MỀM



- 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 với sự xuất hiện bộ phát triển Turbo Pascal đã hình thành một khuynh hướng mới về việc tạo ra các môi trường phát triển tích hợp IDE (Integrated Development Environment) mà toàn bộ các quá trình soạn thảo, dịch, liên kết, thi hành và gỡ lỗi được thực hiện trong cùng một môi trường liên hệ chặt chẽ
- Một bước phát triển tiếp của IDE là việc phát triển hướng đối tượng, phát triển theo mẫu, lập trình hướng tới thành phần (liên kết động các thành phần có sẵn trong mã nhị phân) làm việc sinh mã chương trình trở nên hiệu quả hơn rất nhiều.
- 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 khác.

TỔNG KẾT

- Ngôn ngữ lập trình là phương tiện diễn tả thuật toán để máy tính có thể sử dụng trực tiếp hoặc gián tiếp.
- Theo mức trừu tượng hoá có các mức là ngôn ngữ máy, hợp ngữ và ngôn ngữ thuật toán. Đối với hợp ngữ phải sử dụng phần mềm hợp dịch, với ngôn ngữ thuật toán phải dùng phần mềm biên dịch để tạo ra phần mềm tương ứng trong ngôn ngữ máy – ngôn ngữ mà máy có thể chạy trực tiếp.
- Các bước chính để dịch từ một chương trình nguồn sang mã nhị phân là soạn thảo, phân tích từ vựng, phân tích cú pháp, dịch, tối ưu hoá, liên kết mã. Trong các môi trường tích hợp các khâu trên và cả khâu gỡ lỗi được tích hợp vào trong một tổng thể.

CÂU HỎI VÀ BÀI TẬP

1. So sánh ngôn ngữ thuật toán với ngôn ngữ máy và hợp ngữ
2. Kể tên một số ngôn ngữ lập trình mà bạn biết
3. Nếu các bước thực hiện một chương trình trên ngôn ngữ thuật giải
4. Phân biệt lỗi cú pháp và lỗi ngữ nghĩa
5. Trình bày môi trường phát triển tích hợp

CẢM ƠN ĐÃ THEO DÕI



HẾT BÀI 9. HỎI VÀ ĐÁP

