TIN HỌC LỚP 11

Bài 1: Khái niệm về lập trình và ngôn ngữ

1. Khái niệm lập trình, ngôn ngữ lập trình

* Lập trình là sử dụng cấu trúc dữ liệu và các câu lệnh của một ngôn ngữ lập trình cụ thể để mô phỏng dữ liệu và diễn đạt các thao tác của thuật toán
* Ngôn ngữ lập trình là ngôn ngữ dùng để viết chương trình

2. Phân loại ngôn ngữ lập trình

Ngôn ngữ lập trình được chia thành ba loại:

* Ngôn ngữ máy: Ngôn ngữ duy nhất máy có thể trực tiếp hiểu và thực hiện
* Hợp ngữ: Rất gần với ngôn ngữ máy, nhưng mã lệnh được thay bằng tên viết tắt của thao tác (thường là tiếng Anh)
* Ngôn ngữ bậc cao: Gần với ngôn ngữ tự nhiên, có tính độc lập cao, ít phụ thuộc vào loại máy và chương trình phải dịch sang ngôn ngữ máy mới thực hiện được

3. Chương trình dịch

a. Khái niệm

Chương trình có chức năng chuyển đổi chương trình viết trên ngôn ngữ lập trình bậc cao thành chương trình thực hiện được trên máy tính gọi là chương trình dịch



Hình 1. Chương trình dịch

Trong đó:

* Chương trình nguồn: Là chương trình viết trên ngôn ngữ lập trình bậc cao
* Chương trình đích: Là chương trình nguồn được chuyển đổi sang ngôn ngữ máy nhờ chương trình dịch

b. Phân loại

Chương trình dịch được chia thành hai loại: Thông dịch và biên dịch

b. 1 Thông dịch

* Thực hiện lặp đi lặp lại dãy các bước sau:
  + Kiểm tra tính đúng đắn của câu lệnh tiếp theo trong chương trình nguồn
  + Chuyển đổi câu lệnh đó thành các câu lệnh tương ứng trong ngôn ngữ máy
  + Thực hiện các câu lệnh vừa chuyển đổi
* Loại chương trình dịch này đặc biệt thích hợp cho môi trường đối thoại giữa người dùng và hệ thống, như các ngôn ngữ khai thác hệ quản trị cơ sở dữ liệu, đối thoại với hệ điều hành…

b.2 Biên dịch

* Thực hiện qua hai bước sau:
  + Duyệt, kiểm tra, phát hiện lỗi, xác định chương trình nguồn có dịch được không
  + Dịch chương trình nguồn thành một chương trình đích có thể thực hiện trên máy và  lưu trữ lại để sử dụng về sau
* Loại chương trình dịch này thuận tiện cho các chương trình ổn định và cần thực hiện nhiều lần
* Kèm với chương trình dịch, người dùng còn được cung cấp các dịch vụ lên quan như biên soạn, lưu trữ… tạo thành một môi trường làm việc trên một ngôn ngữ lập trình. Ví dụ: Turbo Pascal 7.0, Free Pascal 1.2, Visual Pascal 2.1,…

Bai2: Các thành phần của ngôn ngữ lập trình

1. Các thành phần cơ bản

Mỗi ngôn ngữ lập trình có 3 thành phần cơ bản là Bảng chữ cái, cú pháp và ngữ nghĩa.

a. Bảng chữ cái

* Tập hợp kí tự được dùng để viết chương trình

Ví dụ:

Trong Pascal bảng chữ cái bao gồm:

* Chữ cái thường và chữ cái in hoa tiếng Anh: a…z ; A..Z
* 10 chữ thập phân: 0..9
* Các kí tự đặc biệt:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ++ | −− | ∗∗ | ╱╱ | == | | << | >> | [[ | ]] | .. | ,, |
| ;; | ## | ∧∧ | αα | $$ | && | (( | )) | {{ | }} | :: | ‘‘ |
| Dấu cách (mã ASCII là 32) | | | | | | \_ Dấu gạch dưới | | | | | |

Bảng 1. Các kí tự đặc biệt trong Pascal

\* Lưu ý:

* Các ngôn ngữ lập trình khác nhau cũng có sự khác nhau về bảng chữ cái
  + Ví dụ: Bảng chữ cái của ngôn ngữ C/C++ so với PASCAL có bổ sung thêm một số kí tự như:   ““ ╲╲ !! ?? %% ||
* Không được phép dùng bất kì kí tự nào ngoài các kí tự quy định trong bảng chữ cái khi viết chương trình

b. Cú pháp

* Là bộ quy tắc để viết chương trình, gồm những quy định viết từ và tổ hợp từ của mỗi ngôn ngữ
* Dựa vào cú pháp người lập trình và chương trình dịch biết tổ hợp nào của các kí tự trong bảng chữ cái là hợp lệ, nhờ đó có thể mô tả chính xác thuật toán để máy thực hiện

c. Ngữ nghĩa

* Xác định ý nghĩa thao tác cần thực hiện, ứng với tổ hợp kí tự dựa vào ngữ cảnh của nó
* Ngữ nghĩa xác định tính chất và thuộc tính của các tổ hợp kí tự tạo thành các dòng lệnh trong chương trình

2. Một số khái niệm

a. Tên

* Dùng để xác định các đối tượng trong chương trình
* Tên đặt theo quy tắc của ngôn ngữ lập trình và từng chương trình dịch cụ thể

|  |  |
| --- | --- |
| Turbo Pascal | C++ |
| * Tên là một dãy liên tiếp các kí tự gồm: chữ cái, chữ số, dấu gạch dưới * Độ dài tên ≤ 127 kí tự * Bắt đầu tên bằng chữ cái hoặc dấu gạch dưới * Không phân biệt chữ hoa và thường | * Độ dài tên tuỳ ý      * Có phân biệt chữ hoa và thường |

Bảng 2. Qui tắc đặt tên trong ngôn ngữ lập trình Pascal và ngôn ngữ lập trình C++

Ví dụ:

* Tên đúng:

LOP11C2, LOP\_11C2

* Tên sai:

LOP 11C2, 3D, A\B

Nhiều ngôn ngữ lập trình phân biệt ba loại tên:

* Tên dành riêng (Từ khoá): là những tên được ngôn ngữ lập trình dùng với ý nghĩa xác định mà không được dùng với ý nghĩa khác
* Tên chuẩn: dùng với ý nghĩa xác định nào đó được quy định trong các thư viện của ngôn ngữ lập trình, nhưng người lập trình có thể khai báo và dùng với ý nghĩa khác
* Tên do người lập trình đặt: sử dụng theo ý nghĩa riêng, xác định bằng cách khai báo trước khi sử dụng

Ví dụ:

|  |  |  |
| --- | --- | --- |
| LOẠI TÊN | PASCAL | C/ C++ |
| Tên dành riêng | PROGRAM, USE, VAR, BEGIN, END,… | MAIN, INCLUDE, VOID, WHILE, IF… |
| Tên chuẩn | BYTE, REAL, ABS… | COUT, CLRSCR, CIN… |
| Tên do người lập trình đặt | BAITAP, A, X1, CHUVI, SO\_LUONG, … | |

Bảng 3. Phân biệt các loại tên trong ngôn ngữ lập trình Pascal và ngôn ngữ lập trình C++

b. Hằng và biến

b.1. Hằng

* Hằng là đại lượng có giá trị không thay đổi trong quá trình thực hiện chương trình
* Hằng số học là các số nguyên và số thực (dấu phẩy tĩnh hay dấu phẩy động)
* Hằng lôgic là các giá trị TRUE hoặc FALSE
* Hằng xâu là chuỗi kí tự bất kì, khi viết đặt trong cặp dấu nháy

|  |  |  |
| --- | --- | --- |
| LOẠI HẰNG | PASCAL | C/ C++ |
| Hằng số học | 3          0           -8           +15  2.5         5.0          -12.79        +6.8           0.2  -2.259E02                  1.7E-3 | |
| Hằng lôgic | TRUE              FALSE | |
| Hằng xâu | ‘Tin hoc’  ‘12345’ | “Tin hoc”  “12345” |

Ví dụ:

Bảng 4. Các loại hằng trong ngôn ngữ lập trình Pascal và ngôn ngữ lập trình C++

b.2. Biến

* Là những đại lượng được đặt tên, dùng để lưu trữ giá trị và giá trị có thể được thay đổi trong quá trình thực hiện chương trình
* Tên biến mang giá trị của biến tại từng thời điểm thực hiện chương trình

c. Chú thích

* Các đoạn chú thích đặt trong chương trình nguồn giúp người đọc dễ dàng nhận biết ý nghĩa của chương trình đó
* Chú thích không làm ảnh hưởng đến nội dung chương trình nguồn và được chương trình dịch bỏ qua
* Trong Pascal, chú thích được đặt giữa cặp dấu {  } hoặc (\*  \*)
* Trong C++, chú thích được đặt giữa cặp dấu /\*  và \*/