

**BỘ LAO ĐỘNG - THƯƠNG BINH VÀ XÃ HỘI
TỔNG CỤC DẠY NGHỀ**

**GIÁO TRÌNH
Môn học: LẬP TRÌNH C
NGHỀ: QUẢN TRỊ MẠNG
TRÌNH ĐỘ: CAO ĐẲNG NGHỀ**

*(Ban hành kèm theo Quyết định số: 120/QĐ-TCDN ngày 25 tháng 02 năm 2013 của Tổng
cục trưởng Tổng cục dạy nghề)*



Hà Nội, năm 2013

TUYÊN BỐ BẢN QUYỀN:

Tài liệu này thuộc loại sách giáo trình nên các nguồn thông tin có thể được phép dùng nguyên bản hoặc trích dùng cho các mục đích về đào tạo và tham khảo.

Mọi mục đích khác mang tính lệch lạc hoặc sử dụng với mục đích kinh doanh thiếu lành mạnh sẽ bị nghiêm cấm.

MÃ TÀI LIỆU: MH18

TaiLieu.vn

LỜI GIỚI THIỆU

Ngày nay, khoa học máy tính thâm nhập vào mọi lĩnh vực. Tự động hóa hiện đang là ngành chủ chốt điều hướng sự phát triển thế giới. Bất cứ ngành nghề nào cũng cần phải hiểu biết ít nhiều về Công nghệ Thông tin và lập trình nói chung. Cụ thể, C là một ngôn ngữ lập trình cấp cao mà mọi lập trình viên cần phải biết. Vì thế, trong giáo trình này, chúng ta sẽ nghiên cứu chi tiết cấu trúc ngôn ngữ C.

Môn học này là nền tảng để tiếp thu hầu hết các môn học khác trong chương trình đào tạo. Mặt khác, nắm vững ngôn ngữ C là cơ sở để phát triển các ứng dụng.

Học xong môn này, sinh viên phải nắm được các vấn đề sau:

- Khái niệm về ngôn ngữ lập trình.
- Khái niệm về kiểu dữ liệu
- Kiểu dữ liệu có cấu trúc (cấu trúc dữ liệu).
- Khái niệm về giải thuật
- Ngôn ngữ biểu diễn giải thuật.
- Ngôn ngữ sơ đồ (lưu đồ), sử dụng lưu đồ để biểu diễn các giải thuật.
- Tổng quan về Ngôn ngữ lập trình C.
- Các kiểu dữ liệu trong C.
- Các lệnh có cấu trúc.
- Cách thiết kế và sử dụng các hàm trong C.
- Một số cấu trúc dữ liệu trong C.

Hà Nội, ngày 25 tháng 02 năm 2013

Tham gia biên soạn

1. Chủ biên Trần Thị Hà Khuê

2. Thành viên Võ Thị Ngọc Tú

3. Thành viên Dương Hiền Tú

MỤC LỤC

LỜI GIỚI THIỆU	3
MỤC LỤC	4
1. Vị trí, tính chất, ý nghĩa và vai trò môn học:.....	7
2. Mục tiêu của môn học:	7
3. Nội dung môn học:	7
CHƯƠNG 1: GIỚI THIỆU VỀ NGÔN NGỮ C	9
1.1. TỔNG QUAN VỀ NGÔN NGỮ LẬP TRÌNH C	9
1.2. KHỞI ĐỘNG VÀ THOÁT KHỎI CHƯƠNG TRÌNH C	11
1.2.1. KHỞI ĐỘNG CHƯƠNG TRÌNH C	11
1.2.2. THOÁT KHỎI CHƯƠNG TRÌNH C	12
1.2.3. CÁC VÍ DỤ ĐƠN GIẢN	12
CHƯƠNG 2: CÁC THÀNH PHẦN CƠ BẢN TRONG NGÔN NGỮ C	14
2.1. Từ khóa	14
2.1.1. Bộ chữ viết trong C	14
2.1.2. Từ khóa	15
2.2. Tên	15
2.3. Kiểu dữ liệu	16
2.3.1. Kiểu số nguyên.....	16
2.3.2. Kiểu số thực	17
2.4. Các phép toán	17
2.4.1. Các phép toán số học.....	17
2.4.2. Các phép toán quan hệ và logic.....	18
2.4.3. Phép toán tăng giảm	20
2.4.4. Thứ tự ưu tiên các phép toán.....	21
2.5. Ghi chú.....	22
2.5.1. Ghi chú	22
2.5.2. Cấu trúc chương trình C	23
2.6. Khai báo biến.....	24
2.6.1. Biến.	24
2.6.2. Vị trí khai báo biến trong C.....	25
2.6.3. Biểu thức	25
2.7. Nhập/xuất dữ liệu	26
2.7.1. Lệnh gán.....	26
2.7.2. Lệnh nhập.....	27

2.7.3. Lệnh xuất.....	28
2.8. Bài tập thực hành	29
CHƯƠNG 3: CẤU TRÚC Rẽ NHÁNH CÓ ĐIỀU KIỆN	30
3.1. Lệnh và khối lệnh	30
3.1.1. Lệnh.....	30
3.1.2. Khối lệnh	30
3.2. Lệnh if.....	31
3.2.1. Dạng 1 (if thiếu)	31
3.2.2. Bài tập thực hành.....	32
3.2.3. Dạng 2 (if đủ)	32
3.2.4. Bài tập thực hành.....	33
3.2.5. Cấu trúc else if	33
3.2.6. Bài tập thực hành.....	35
3.2.7. Cấu trúc if lồng nhau	35
3.2.8. Bài tập thực hành.....	37
3.3. Lệnh switch()	37
3.3.1. Cấu trúc switch...case (switch thiếu).....	37
3.3.2. Bài tập thực hành.....	39
3.3.3. Cấu trúc switch...case...default (switch đủ).....	40
3.3.4. Bài tập thực hành.....	42
3.3.5. Cấu trúc switch lồng.....	42
3.3.6. Bài tập thực hành.....	44
CHƯƠNG 4: CẤU TRÚC VÒNG LẶP	45
4.1. Lệnh for	45
4.2. Lệnh break	48
4.3. Lệnh continue	49
4.4. Lệnh while	50
4.5. Lệnh do..while	51
4.6. Vòng lặp lồng nhau.....	53
4.7. So sánh sự khác nhau của các vòng lặp.....	54
4.8. Bài tập thực hành	54
CHƯƠNG 5: HÀM.....	55
5.1. Các ví dụ về hàm	55
5.1.1. Khái niệm về hàm	55
5.1.2. Các ví dụ về hàm.....	57
5.2. THAM SỐ DẠNG THAM BIẾN VÀ THAM TRỊ.....	61
5.2.1. Tham số dạng tham trị.....	61
5.2.2. Bài tập thực hành về tham trị	63
5.2.3. Tham số dạng tham biến	63

5.2.4. Bài tập thực hành.....	64
5.3. SỬ DỤNG BIẾN TOÀN CỤC.....	65
5.3.1. Sử dụng biến toàn cục	65
5.3.2. Bài tập thực hành.....	67
5.4. Dùng dẫn hướng #define	70
CHƯƠNG 6: MẢNG VÀ CHUỖI	72
6.1. GIỚI THIỆU KIỂU DỮ LIỆU “KIỂU MẢNG” TRONG C	72
6.2. MẢNG MỘT CHIỀU	73
6.2.1. Khai báo	73
6.2.2. Truy xuất từng phần tử của mảng	74
6.2.3. Bài tập thực hành.....	77
6.3. MẢNG NHIỀU CHIỀU	78
6.3.1. Khai báo	79
6.3.2. Truy xuất từng phần tử của mảng hai chiều.....	80
6.3.3. Bài tập thực hành.....	82
6.4. CHUỖI.....	83
6.4.1. KHÁI NIỆM.....	84
6.4.2. KHAI BÁO	84
6.4.3. CÁC THAO TÁC TRÊN CHUỖI KÝ TỰ	85
6.4.4. Bài tập thực hành.....	90
TÀI LIỆU THAM KHẢO	93

MÔN HỌC LẬP TRÌNH C

Mã môn học: MH18

1. Vị trí, tính chất, ý nghĩa và vai trò môn học:

- Vị trí: Môn học được bố trí sau khi sinh viên học xong các môn học chung, các môn học cơ sở chuyên ngành đào tạo chuyên môn nghề.
- Tính chất: Lập trình C là môn học lý thuyết cơ sở nghề.
- Ý nghĩa và vai trò: Môn học này là nền tảng để tiếp thu hầu hết các môn học khác trong chương trình đào tạo. Mặt khác, nắm vững ngôn ngữ C là cơ sở để phát triển các ứng dụng.

2. Mục tiêu của môn học:

- Trình bày được công dụng của ngôn ngữ lập trình C;
- Trình bày được cú pháp, công dụng của các câu lệnh trong ngôn ngữ C;
- Phân tích được chương trình: xác định nhiệm vụ chương trình (phải làm gì?).
- Viết chương trình và thực hiện chương trình đơn giản trong máy tính bằng ngôn ngữ C.
- Bố trí làm việc khoa học đảm bảo an toàn cho người và phương tiện học tập.

3. Nội dung môn học:

Số TT	Tên chương, mục	Thời gian			
		Tổng số	Lý thuyết	Thực hành Bài tập	Kiểm tra* (LT hoặc TH)
I	Giới thiệu về ngôn ngữ C	1	1		
II	Các thành phần trong ngôn ngữ C Từ khóa Tên Kiểu dữ liệu Ghi chú Khai báo biến Nhập/xuất dữ liệu	7	6	1	
III	Cấu trúc rẽ nhánh có điều kiện Lệnh và khối lệnh Lệnh if Lệnh switch	12	5	6	1
IV	Cấu trúc vòng lặp Lệnh for Lệnh Break Lệnh continue	12	6	6	

Số TT	Tên chương, mục	Thời gian			
		Tổng số	Lý thuyết	Thực hành Bài tập	Kiểm tra* (LT hoặc TH)
V	Lệnh while Lệnh do...while Vòng lặp lồng nhau So sánh sự khác nhau của các vòng lặp	13	5	7	1
	Hàm Các ví dụ về hàm Tham số dạng tham biến và tham trị Sử dụng biến toàn cục Dùng dẫn hướng #define				
VI	Mảng và chuỗi Mảng Chuỗi	15	7	7	1
	Cộng	60	30	27	3

CHƯƠNG 1: GIỚI THIỆU VỀ NGÔN NGỮ C

Mã chương: MH18-01

Ý nghĩa:

C là ngôn ngữ rất có hiệu quả và được ưa chuộng nhất để viết các phần mềm hệ thống, mặc dù nó cũng được dùng cho việc viết các ứng dụng. Ngoài ra, C cũng thường được dùng làm phương tiện giảng dạy trong khoa học máy tính mặc dù ngôn ngữ này không được thiết kế dành cho người nhập môn.

Mục tiêu:

- Trình bày được lịch sử phát triển của ngôn ngữ C;
- Mô tả được những ứng dụng thực tế của ngôn ngữ C.
- Một số thao tác cơ bản của trình soạn thảo C.
- Tiếp cận một số lệnh đơn giản thông qua các ví dụ
- Thực hiện các thao tác an toàn với máy tính.

Nội dung:

1.1. TỔNG QUAN VỀ NGÔN NGỮ LẬP TRÌNH C

Mục tiêu:

- *Trình bày được lịch sử phát triển của ngôn ngữ C;*
- *Mô tả được những ứng dụng thực tế của ngôn ngữ C.*
- *Mô tả những đặc điểm cơ bản của ngôn ngữ C*

C là ngôn ngữ lập trình cấp cao, được sử dụng rất phổ biến để lập trình hệ thống cùng với Assembler và phát triển các ứng dụng.

Vào những năm cuối thập kỷ 60 đầu thập kỷ 70 của thế kỷ XX, Dennis Ritchie (làm việc tại phòng thí nghiệm Bell) đã phát triển ngôn ngữ lập trình C dựa trên ngôn ngữ BCPL (do Martin Richards đưa ra vào năm 1967) và ngôn ngữ B (do Ken Thompson phát triển từ ngôn ngữ BCPL vào năm 1970 khi viết hệ điều hành UNIX đầu tiên trên máy PDP-7) và được cài đặt lần đầu tiên trên hệ điều hành UNIX của máy DEC PDP-11. Năm 1978, Dennis Ritchie và B.W Kernighan đã cho xuất bản quyển “Ngôn ngữ lập trình C” và được phổ biến rộng rãi đến nay.

Lúc ban đầu, C được thiết kế nhằm lập trình trong môi trường của hệ điều hành Unix nhằm mục đích hỗ trợ cho các công việc lập trình phức tạp. Nhưng về sau, với những nhu cầu phát triển ngày một tăng của công việc lập trình, C đã vượt qua khuôn khổ của phòng thí nghiệm Bell và nhanh chóng hội nhập vào thế giới lập trình để rồi các công ty lập trình sử dụng một cách rộng rãi. Sau đó, các công ty sản xuất phần mềm lần lượt đưa ra các phiên bản hỗ trợ cho việc lập trình bằng ngôn ngữ C và chuẩn ANSI C cũng được khai sinh từ đó.

Ngôn ngữ lập trình C là một ngôn ngữ lập trình hệ thống rất mạnh và rất “mềm dẻo”, có một thư viện gồm rất nhiều các hàm (function) đã được tạo sẵn. Người lập trình có thể tận dụng các hàm này để giải quyết các bài toán mà không cần phải tạo mới. Hơn thế nữa, ngôn ngữ C hỗ trợ rất nhiều phép toán nên phù hợp cho việc giải quyết các bài toán kỹ thuật có nhiều công thức phức tạp. Ngoài ra, C cũng cho phép người lập trình tự định nghĩa thêm các kiểu dữ liệu trừu tượng khác. Tuy nhiên, điều mà người mới vừa học lập trình C thường gặp “rắc rối” là “hơi khó hiểu” do sự “mềm dẻo” của C. Dù vậy, C được phổ biến khá rộng rãi và đã trở thành một công cụ lập trình khá mạnh, được sử dụng như là một ngôn ngữ lập trình chủ yếu trong việc xây dựng những phần mềm hiện nay.

Ngôn ngữ C có những đặc điểm cơ bản sau:

- o Tính cô đọng (compact): C chỉ có 32 từ khóa chuẩn và 40 toán tử chuẩn, nhưng hầu hết đều được biểu diễn bằng những chuỗi ký tự ngắn gọn.

- o Tính cấu trúc (structured): C có một tập hợp những chỉ thị của lập trình như cấu trúc lựa chọn, lặp... Từ đó các chương trình viết bằng C được tổ chức rõ ràng, dễ hiểu.

- o 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.

o Tính linh động (flexible): C là một ngôn ngữ rất uyển chuyển và cú pháp, chấp nhận nhiều cách thể hiện, có thể thu gọn kích thước của các mã lệnh làm chương trình chạy nhanh hơn.

o Biên dịch (compile): C 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.

1.2. KHỞI ĐỘNG VÀ THOÁT KHỎI CHƯƠNG TRÌNH C

Mục tiêu:

- Một số thao tác cơ bản của trình soạn thảo C.
- Tiếp cận một số lệnh đơn giản thông qua các ví dụ
- Thực hiện các thao tác an toàn với máy tính.

1.2.1. KHỞI ĐỘNG CHƯƠNG TRÌNH C

Nhập lệnh tại dấu nhắc DOS: gõ BC ↵ (Enter) (nếu đường dẫn đã được cài bằng lệnh path trong đó có chứa đường dẫn đến thư mục chứa tập tin BC.EXE). Nếu đường dẫn chưa được cài đặt ta tìm xem thư mục BORLAND C (hoặc TURBO C) nằm trong ổ đĩa nào. Sau đó ta gõ lệnh sau:

<ổ đĩa>:\BORLAND C\BIN\BC↵(Enter)

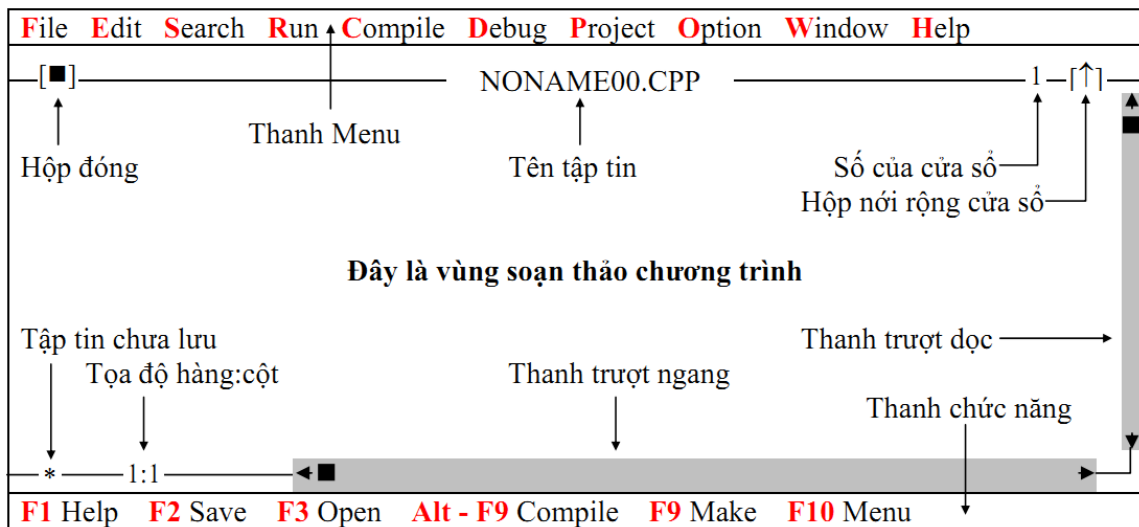
Nếu vừa khởi động BC vừa soạn thảo chương trình với một tập tin có tên do chúng ta đặt, thì gõ lệnh: BC [đường dẫn]<tên file cần soạn thảo>, nếu tên file cần soạn thảo đã có thì được nạp lên, nếu chưa có sẽ được tạo mới.

Khởi động tại Windows: Bạn vào menu Start, chọn Run, bạn gõ vào hộp Open 1 trong các dòng lệnh như nhập tại DOS. Hoặc bạn vào Window Explorer, chọn ổ đĩa chứa thư mục BORLANDC, vào thư mục BORLANDC, vào thư mục BIN, khởi động tập tin BC.EXE.

Ví dụ: gõ D:\BORLANDC\BIN\BIN\BC E:\BAITAP_BC\VIDU1.CPP

Câu lệnh trên có nghĩa khởi BC và nạp tập tin VIDU1.CPP chứa trong thư mục BAITAP_BC trong ổ đĩa E. Nếu tập tin này không có sẽ được tạo mới.

Màn hình sau khi khởi động thành công:



Hình 1.1. Màn hình sau khi khởi động C thành công

1.2.2. THOÁT KHỎI CHƯƠNG TRÌNH C

Ấn phím F10 (kích hoạt Menu), chọn menu File, chọn Quit;

Hoặc ấn tổ hợp phím Alt – X

1.2.3. CÁC VÍ DỤ ĐƠN GIẢN

```
1  /* Chuong trinh in ra cau bai hoc C dau tien
   */
2  #include<stdio.h>
3
4  void main()
5  {
6      printf("Bai hoc C dau tien");
7  }
```

Kết quả in ra màn hình:

Bai hoc C dau tien

Dòng thứ 1: bắt đầu bằng `/*` và kết thúc bằng `*/` cho biết hàng này là hàng diễn giải (chú thích). Khi dịch và chạy chương trình, dòng này không được dịch và cũng không thi hành lệnh gì cả. Mục đích của việc ghi chú này giúp chương trình rõ ràng hơn. Sau này chúng ta đọc lại chương trình biết chương trình làm gì.

Dòng thứ 2: chứa phát biểu tiền xử lý `#include <stdio.h>`. Vì trong chương trình này chúng ta sử dụng hàm thư viện của C là `printf`, do đó chúng ta cần phải

có khai báo của hàm thư viện này để báo cho trình biên dịch C biết. Nếu không khai báo chương trình sẽ báo lỗi.

Dòng thứ 3: Hàng trắng viết ra với ý đồ làm cho bảng chương trình thoáng, dễ đọc.

Dòng thứ 4: `void main(void)` là thành phần chính của mọi chương trình C (bạn có thể viết `main()` hoặc `void main()` hoặc `main(void)`). Tuy nhiên, bạn nên viết theo dạng `void main(void)` để chương trình rõ ràng hơn. Mọi chương trình C đều bắt đầu thi hành từ hàm `main`. Cặp dấu ngoặc `()` cho biết đây là khối hàm (function). Hàm `void main(void)` có từ khóa `void` đầu tiên cho biết hàm này không trả về giá trị, từ khóa `void` trong ngoặc đơn cho biết hàm này không nhận vào đối số.

Dòng thứ 5 và 7: cặp dấu ngoặc móc `{ }` giới hạn thân của hàm. Thân hàm bắt đầu bằng dấu `{` và kết thúc bằng dấu `}`.

Dòng thứ 6: `printf("Bai hoc C dau tien.");`, chỉ thị cho máy in ra chuỗi ký tự nằm trong nháy kép (`"`). Hàng này được gọi là một câu lệnh, kết thúc một câu lệnh trong C phải là dấu chấm phẩy (`;`).

Chú ý:

- Các từ `include`, `stdio.h`, `void`, `main`, `print` phải viết bằng chữ thường.
- Chuỗi trong nháy kép cần in ra "Bạn có thể viết chữ HOA, thường tùy ý".
- Kết thúc câu lệnh phải có dấu chấm phẩy
- Kết thúc tên hàm không có dấu chấm phẩy hoặc bất cứ dấu gì.
- Ghi chú phải đặt trong cặp `/*.....*/`
- Thân hàm phải được bao bởi cặp `{ }`.
- Các câu lệnh trong thân hàm phải viết thụt vào.

Sau khi nhập xong đoạn chương trình vào máy. Bạn ấn và giữ phím `Ctrl`, gõ `F9` để dịch và chạy chương trình. Khi đó chúng ta thấy chương trình chớp rất nhanh và không thấy kết quả gì cả. Bạn Ấn và giữ phím `Alt`, gõ `F5` để xem kết quả, khi xem xong, ấn phím bất kỳ để quay về màn hình soạn thảo chương trình.

CHƯƠNG 2: CÁC THÀNH PHẦN CƠ BẢN TRONG NGÔN NGỮ C

Mã chương: MH18-02

Ý nghĩa:

Ngôn ngữ C hỗ trợ rất nhiều phép toán nên phù hợp cho việc giải quyết các bài toán kỹ thuật có nhiều công thức phức tạp. Ngoài ra, C cũng cho phép người lập trình tự định nghĩa thêm các kiểu dữ liệu trừu tượng khác.

Mục tiêu:

- Sử dụng được hệ thống kí hiệu và từ khóa
- Khai báo tên đúng
- Trình bày được các kiểu dữ liệu
- Khai báo biến đúng
- Thực hiện được việc nhập và xuất dữ liệu
- Thực hiện các thao tác an toàn với máy tính

2.1. Từ khóa

Mục tiêu: Hiểu và sử dụng đúng hệ thống ký hiệu và từ khóa.

2.1.1. Bộ chữ viết trong C

Ngôn ngữ C được xây dựng trên bộ ký tự sau:

- 26 chữ cái hoa: A B C .. Z
- 26 chữ cái thường: a b c .. z
- 10 chữ số: 0 1 2 .. 9
- Các ký hiệu toán học: + - * / = ()
- Ký tự gạch nối: _
- Các ký tự khác: . , ; [] { } ! \ & % # \$...
- Dấu cách (space) dùng để tách các từ. Ví dụ chữ SINH VIEN có 9 ký tự, còn SINHVIEN chỉ có 8 ký tự.

* Chú ý:

Khi viết chương trình, ta không được sử dụng bất kỳ ký tự nào khác ngoài các ký tự trên.

2.1.2. Từ khoá

Từ khoá là những từ được sử dụng để khai báo các kiểu dữ liệu, để viết các toán tử và các câu lệnh. Một số từ khoá của TURBO C được liệt kê như sau:

asm	break	case	cdecl
char	const	continue	default
do	double	else	enum
extern	Far	float	for
goto	Huge	if	int
interrupt	Long	near	pascal
register	return	short	signed
sizeof	static	struct	switch
typedef	union	unsigned	void
volatile	while		

Chú ý:

- Không được dùng các từ khoá để đặt tên cho các hằng, biến, mảng, hàm ...
- Từ khoá phải được viết bằng chữ thường, ví dụ: viết từ khoá khai báo kiểu nguyên là int chứ không phải là INT.

2.2. Tên

Mục tiêu: Hiểu và khai báo tên theo đúng nguyên tắc.

Khái niệm tên rất quan trọng trong quá trình lập trình, nó không những thể hiện rõ ý nghĩa trong chương trình mà còn dùng để xác định các đại lượng khác nhau khi thực hiện chương trình.

Tên thường được đặt cho hằng, biến, mảng, con trỏ, nhãn... Chiều dài tối đa của tên là 32 ký tự.

Tên biến hợp lệ là một chuỗi ký tự liên tục gồm: Ký tự chữ, số và dấu gạch dưới. Ký tự đầu của tên phải là chữ hoặc dấu gạch dưới. Khi đặt tên không được đặt trùng với các từ khóa.

Ví dụ 1 :

Các tên đúng: delta, a_1, Num_ODD, Case

Các tên sai:

3a_1 (ký tự đầu là số)

num-odd (sử dụng dấu gạch ngang)

int (đặt tên trùng với từ khóa)

del ta (có khoảng trắng)

f(x) (có dấu ngoặc tròn)

Lưu ý: Trong C, tên phân biệt chữ hoa, chữ thường

Ví dụ 2 : number khác Number

case khác Case

(case là từ khóa, do đó bạn đặt tên là Case vẫn đúng)

2.3. Kiểu dữ liệu

Mục tiêu: Hiểu và có thể sử dụng kiểu dữ liệu theo đúng mục đích của bài toán.

2.3.1. Kiểu số nguyên

Trong C cho phép sử dụng số nguyên kiểu int, số nguyên dài kiểu long và số nguyên không dấu kiểu unsigned. Kích cỡ và phạm vi biểu diễn của chúng được chỉ ra trong bảng dưới đây:

Kiểu	Phạm vi biểu diễn	Kích thước
------	-------------------	------------

int	-32768 đến 32767	2 byte
unsigned int	0 đến 65535	2 byte
long	-2147483648 đến 2147483647	4 byte
unsigned long	0 đến 4294967295	4 byte

Bảng 2.1. Kích cỡ và phạm vi biểu diễn kiểu dữ liệu số nguyên

2.3.2. Kiểu số thực

Trong C cho phép sử dụng số thực gồm 3 kiểu: float, double và long double. Kích cỡ và phạm vi biểu diễn của chúng được chỉ ra trong bảng dưới đây:

Kiểu	Phạm vi biểu diễn	Số chữ số có nghĩa	Kích thước
Float	3.4E-38 đến 3.4E+38	7 đến 8	4 byte
Double	1.7E-308 đến 1.7E+308	15 đến 16	8 byte
long double	3.4E-4932 đến 1.1E4932	17 đến 18	10 byte

Bảng 2.2. Kích cỡ và phạm vi biểu diễn kiểu số thực

Giải thích:

Máy tính có thể lưu trữ được các số kiểu float có giá trị tuyệt đối từ 3.4E-38 đến 3.4E+38. Các số có giá trị tuyệt đối nhỏ hơn 3.4E-38 được xem bằng 0. Phạm vi biểu diễn của số double được hiểu theo nghĩa tương tự.

2.4. Các phép toán

Mục tiêu : Hiểu và có thể thực hiện các phép tính toán trong chương trình.

2.4.1. Các phép toán số học

Phép toán	Ý nghĩa	Ví dụ
-----------	---------	-------

Phép toán	Ý nghĩa	Ví dụ
+	Phép cộng	$a+b$
-	Phép trừ	$a-b$
*	Phép nhân	$a*b$
/	Phép chia	a/b (Chia số nguyên sẽ chắt phần thập phân)
%	Phép lấy phần dư	$a\%b$ (Cho phần dư của phép chia a cho b)

Bảng 2.3. Các phép toán số học

Có phép toán một ngôi - ví dụ $-(a+b)$ sẽ đảo giá trị của phép cộng $(a+b)$.

Ví dụ:

$$11/3=3$$

$$11\%3=2$$

$$-(2+6)=-8$$

Các phép toán $+$ và $-$ có cùng thứ tự ưu tiên, có thứ tự ưu tiên nhỏ hơn các phép $*$, $/$, $\%$ và cả ba phép này lại có thứ tự ưu tiên nhỏ hơn phép trừ một ngôi.

Các phép toán số học được thực hiện từ trái sang phải. Số ưu tiên và khả năng kết hợp của phép toán được chỉ ra trong một mục sau này.

2.4.2. Các phép toán quan hệ và logic

Phép toán quan hệ và logic cho ta giá trị đúng (1) hoặc giá trị sai (0). Nói cách khác, khi các điều kiện nêu ra là đúng thì ta nhận được giá trị 1, trái lại ta nhận giá trị 0.

Các phép toán quan hệ:

Phép toán

Ý nghĩa

Ví dụ

>	So sánh lớn hơn	$a > b$ $4 > 5$ có giá trị 0
>=	So sánh lớn hơn hoặc bằng	$a \geq b$ $6 \geq 2$ có giá trị 1
<	So sánh nhỏ hơn	$a < b$ $6 < 7$ có giá trị 1
<=	So sánh nhỏ hơn hoặc bằng	$a \leq b$ $8 \leq 5$ có giá trị 0
==	So sánh bằng nhau	$a == b$ $6 == 6$ có giá trị 1
!=	So sánh khác nhau	$a != b$ $9 != 9$ có giá trị 0

Bốn phép toán đầu có cùng số ưu tiên, hai phép sau có cùng số thứ tự ưu tiên nhưng thấp hơn số thứ tự của bốn phép đầu.

Các phép toán quan hệ có số thứ tự ưu tiên thấp hơn so với các phép toán số học, cho nên biểu thức: $i < n - 1$ được hiểu là $i < (n - 1)$.

2.4.2.1. Các phép toán logic

Trong C sử dụng ba phép toán logic:

Phép phủ định một ngôi (!)

A	!a
khác 0	0
bằng 0	1

Phép và (AND) && - Phép hoặc (OR) ||

a	B	a&&b	a b
khác 0	khác 0	1	1
khác 0	bằng 0	0	1
bằng 0	khác 0	0	1
bằng 0	bằng 0	0	0

Các phép quan hệ có số ưu tiên nhỏ hơn so với ! nhưng lớn hơn so với && và ||, vì vậy biểu thức sau:

$(a < b) \&\& (c > d)$

có thể viết lại thành:

$a < b \&\& c > d$

* Chú ý: Cả a và b có thể là nguyên hoặc thực.

2.4.3. Phép toán tăng giảm

C đưa ra hai phép toán một ngôi để tăng và giảm các biến (nguyên và thực). Toán tử tăng là ++ sẽ cộng 1 vào toán hạng của nó, toán tử giảm -- thì sẽ trừ toán hạng đi 1.

Ví dụ:

n=5

++n Cho ta n=6

--n Cho ta n=4

Ta có thể viết phép toán ++ và -- trước hoặc sau toán hạng như sau: ++n, n++, --n, n--