



GIÁO TRÌNH MÔN HỌC

LẬP TRÌNH C CĂN BẢN

TRÌNH ĐỘ TRUNG CẤP

NGHỀ: TIN HỌC ỨNG DỤNG

Ban hành theo Quyết định số 498/QĐ-CDGTVTTWI-ĐT ngày
25/03/2019 của Hiệu trưởng Trường Cao đẳng GTVT Trung ương I

Hà Nội, 2019

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 dẫn 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: **THUD – MH13**

LỜI NÓI ĐẦU

Trong những năm qua, dạy nghề đã có những bước tiến vượt bậc cả về số lượng và chất lượng, nhằm thực hiện nhiệm vụ đào tạo nguồn nhân lực kỹ thuật trực tiếp đáp ứng nhu cầu xã hội.

Tin học là một ngành khoa học mũi nhọn phát triển hết sức nhanh chóng và chỉ chưa mấy gần đây và ngày càng mở rộng lĩnh vực nghiên cứu, ứng dụng trong mọi mặt của đời sống xã hội

Ngôn ngữ lập trình là một loại công cụ giúp con người thể hiện các vấn đề của thực tế lên máy tính một cách hữu hiệu. Với sự phát triển của Tin học, các ngôn ngữ lập trình cũng dần tiến hóa để đáp ứng các thách thức mới của thực tế

Khoảng cuối những năm 1960 đầu 1970 xuất hiện nhu cầu cần có các ngôn ngữ bậc cao để hỗ trợ cho những nhà Tin học trong việc xây dựng các phần mềm hệ thống, hệ điều hành. Ngôn ngữ C ra đời từ đó, nó đã được phát triển tại phòng thí nghiệm Bell. Đến năm 1978, giáo trình "Ngôn ngữ lập trình C" do chính các tác giả của ngôn ngữ là Dennish Ritchie và B.W.Kerni ghan viết, đã được xuất bản và phổ biến rộng rãi.

C là ngôn ngữ lập trình vạn năng. Ngoài việc C được dùng để viết hệ điều hành UNIX, người ta nhanh chóng nhận ra sức mạnh của C trong việc xử lý cho các vấn đề hiện tại của Tin học. C không gắn với bất kỳ một hệ điều hành hay máy nào, và mặc dầu nó được gọi là "ngôn ngữ lập trình hệ thống" vì nó được dùng cho việc viết hệ điều hành, nó cũng tiện lợi cho cả việc viết các chương trình xử lý số, xử lý văn bản và cơ sở dữ liệu.

Cuốn giáo trình Ngôn ngữ lập trình C dành cho học sinh - sinh viên trình độ Trung cấp, Cao đẳng ngành Tin học ứng dụng của Trường Cao đẳng Giao thông vận tải Trung ương I. Giáo trình được sử dụng làm giáo trình nội bộ trong Nhà trường mà còn được sử dụng làm tài liệu giảng dạy cho giáo viên khi lên lớp và cho học sinh học tập và tham khảo.

Cấu trúc của giáo trình gồm những nội dung sau:

Chương 1: Tổng quan về môn học

Chương 2: Giới thiệu về ngôn ngữ C và môi trường turbo C

Chương 3: Các thành phần trong ngôn ngữ C và nhập xuất dữ liệu

Chương 4: Các lệnh đơn và các lệnh có cấu trúc

Chương 5. Hàm, mảng, chuỗi

Chương 6: con trỏ và các kiểu dữ liệu tự tạo

Khi biên soạn, chúng tôi đã tham khảo các giáo trình và tài liệu giảng dạy môn học này của một số trường đại học trong và ngoài nước để giáo trình vừa

đạt yêu cầu cao về nội dung vừa thích hợp với đối tượng là học sinh - sinh viên ngành Tin học ứng dụng trình độ Cao đẳng, Trung cấp của trường Cao đẳng Giao thông vận tải Trung ương I.

Mặc dù có rất nhiều cố gắng khi biên soạn, nhưng không thể tránh khỏi những sai sót. Rất mong nhận được sự góp ý của các thầy cô và đồng nghiệp để giáo trình hoàn thiện hơn.

Trân trọng cảm ơn!

NHÓM TÁC GIẢ

MỤC LỤC

LỜI NÓI ĐẦU	13
Mục tiêu:	15
Nội dung chương:.....	15
1.1. Giải thuật (Algorithm).....	15
1.2. Chương trình (Program)	15
1.3. Ngôn ngữ lập trình (Programming language)	15
1.4. Ngôn ngữ tự nhiên.....	16
1.5. Ngôn ngữ sơ đồ (lưu đồ).....	16
CHƯƠNG 2: GIỚI THIỆU VỀ NGÔN NGỮ C VÀ MÔI TRƯỜNG TURBO C	19
Mục tiêu:	19
Nội dung chương:.....	19
2.1 Tổng quan về ngôn ngữ lập trình C.	19
2.2 Môi trường làm việc và cách sử dụng Turbo C 3.0.	20
2.2.1 Khởi động và thoát Borland C.	20
2.2.1.1. Khởi động	20
2.2.1.2. Thoát	20
2.2.2 Dịch và Chạy chương trình.....	20
2.2.3. Các ví dụ	21
2.2.3.1. Ví dụ 1	21
2.2.3.2. Ví dụ 2	22
Mục tiêu:	24
Nội dung chương:.....	24
3.1 Từ khóa và tên.....	24
3.1.1. Từ khóa.....	24
3.1.2.Tên	24
3.2. Kiểu dữ liệu.....	25
3.3. Ghi chú trong ngôn ngữ C	25
3.4. Khai báo biến	25

3.4.1. Tên biến	25
3.4.2. Khai báo biến	25
3.4.3. Phạm vi của biến:	25
3.5 Hàm Printf.....	26
3.6 Hàm Scanf.....	27
CHƯƠNG 4: CÁC LỆNH ĐƠN VÀ CÁC LỆNH CÓ CẤU TRÚC	28
Mục tiêu:	28
Nội dung chương:.....	28
4.1 Lệnh và khối lệnh	28
4.1.1 Lệnh	28
4.1.2 Khối lệnh	28
4.2 Lệnh IF.....	28
4.2.1. Lệnh IF dạng 1 (if thiếu)	28
4.2.2 Lệnh IF dạng 2 (if đủ)	31
4.2.3. Cấu trúc lệnh else if	33
4.2.3.1 Cú pháp lệnh:	33
4.2.3.2. Lưu đồ	33
4.2.3.3. Ví dụ:	34
4.2.3.4. Bài tập:	35
4.2.4 Cấu trúc if lồng	36
4.2.4.1. Cú pháp	36
4.2.4.2. Ví dụ	36
4.2.4.3. Bài tập	37
4.3 Lệnh switch	39
4.3.1 Cấu trúc lệnh switch case (switch thiếu)	39
4.3.1.1.Ví dụ: Viết chương trình nhập vào số 1,2,3. In ra tương ứng 1,2,3 sao	40
4.3.1.2. Bài tập	41
4.3.2 Cấu trúc lệnh switch case Default (switch đủ)	42

4.3.2.1. Ví dụ:	44
4.3.2.2. Bài tập	44
4.3.3 Cấu trúc lệnh switch lồng:	45
4.3.3.1. Ví dụ	45
4.3.3.2. Bài tập	47
4.4. Cấu trúc vòng lặp	48
4.4.1 Lệnh For	48
4.4.1.1. Ví dụ:	49
4.4.1.2. Bài tập:	49
4.4.2 Lệnh Break	50
4.4.3 Lệnh Continue	50
4.4.4 Lệnh While	50
4.4.4.1. Ví dụ:	51
4.4.4.2. Bài tập:	52
4.4.5 Lệnh Do While	52
4.4.5.1. Ví dụ:	53
4.4.5.2. Bài tập:	53
4.4.6. Vòng lặp lồng nhau	54
CHƯƠNG 5: HÀM, MẢNG VÀ CHUỖI	56
Mục tiêu:	56
Nội dung chương:	56
5.1. Các ví dụ về hàm	56
5.2. Tham số dạng tham biến và tham trị	59
5.3. Sử dụng biến toàn cục và dùng dẫn hướng #define	60
5.4. Mảng	63
5.4.1 Cách khai báo mảng	63
5.4.2 Tham chiếu đến từng phần tử mảng	64
5.4.3 Nhập dữ liệu cho mảng	64
5.4.4 Sử dụng biến kiểu khác	64

5.3.5 Khởi tạo mảng	64
5.3.6 Khởi tạo mảng không bao hàm kích thước	65
5.5. Chuỗi	65
5.5.1 Cách khai báo chuỗi	65
5.5.2 Hàm nhập (gets) và xuất (puts) chuỗi	66
5.5.3 Khởi tạo chuỗi	67
CHƯƠNG 6: CON TRỎ VÀ CÁC KIỂU DỮ LIỆU TỰ TẠO	68
Mục tiêu:	68
Nội dung chương:	68
6.1. Con trỏ	68
6.1.1. Khai báo biến con trỏ	68
6.1.2 Truyền đại chỉ sang hàm	69
6.1.3 Con trỏ và mảng	70
6.1.4 Con trỏ trỏ đến mảng trong hàm	70
6.1.5 Con trỏ và chuỗi	71
6.2. Structure	72
6.2.1 Khai báo kiểu Structure	72
6.2.2 Cách khai báo biến có kiểu Structure	73
6.2.3 Tham chiếu các phần tử trong Structure	73
6.2.4 Khởi tạo Structure.	74

LỜI NÓI ĐẦU

Trong những năm qua, dạy nghề đã có những bước tiến vượt bậc cả về số lượng và chất lượng, nhằm thực hiện nhiệm vụ đào tạo nguồn nhân lực kỹ thuật trực tiếp đáp ứng nhu cầu xã hội.

Tin học là một ngành khoa học mũi nhọn phát triển hết sức nhanh chóng và chỉ chưa mấy chốc đây và ngày càng mở rộng lĩnh vực nghiên cứu, ứng dụng trong mọi mặt của đời sống xã hội.

Ngôn ngữ lập trình là một loại công cụ giúp con người thể hiện các vấn đề của thực tế lên máy tính một cách hữu hiệu. Với sự phát triển của Tin học, các ngôn ngữ lập trình cũng dần tiến hóa để đáp ứng các thách thức mới của thực tế.

Khoảng cuối những năm 1960 đầu 1970 xuất hiện nhu cầu cần có các ngôn ngữ bậc cao để hỗ trợ cho những nhà Tin học trong việc xây dựng các phần mềm hệ thống, hệ điều hành. Ngôn ngữ C ra đời từ đó, nó đã được phát triển tại phòng thí nghiệm Bell. Đến năm 1978, giáo trình "Ngôn ngữ lập trình C" do chính các tác giả của ngôn ngữ là Dennis Ritchie và B.W.Kernighan viết, đã được xuất bản và phổ biến rộng rãi.

C là ngôn ngữ lập trình vạn năng. Ngoài việc C được dùng để viết hệ điều hành UNIX, người ta nhanh chóng nhận ra sức mạnh của C trong việc xử lý cho các vấn đề hiện tại của Tin học. C không gắn với bất kỳ một hệ điều hành hay máy nào, và mặc dầu nó được gọi là "ngôn ngữ lập trình hệ thống" vì nó được dùng cho việc viết hệ điều hành, nó cũng tiện lợi cho cả việc viết các chương trình xử lý số, xử lý văn bản và cơ sở dữ liệu.

Cuốn giáo trình Ngôn ngữ lập trình C dành cho học sinh - sinh viên trình độ Trung cấp, Cao đẳng ngành Tin học ứng dụng của Trường Cao đẳng Giao thông vận tải Trung ương I. Giáo trình được sử dụng làm giáo trình nội bộ trong Nhà trường mà còn được sử dụng làm tài liệu giảng dạy cho giáo viên khi lên lớp và cho học sinh học tập và tham khảo.

Cấu trúc của giáo trình gồm những nội dung sau:

Chương 1: Tổng quan về môn học

Chương 2: Giới thiệu về ngôn ngữ C và môi trường turbo C

Chương 3: Các thành phần trong ngôn ngữ C và nhập xuất dữ liệu

Chương 4: Các lệnh đơn và các lệnh có cấu trúc

Chương 5: Hàm, mảng, chuỗi

Chương 6: Con trỏ và các kiểu dữ liệu tự tạo

Khi biên soạn, chúng tôi đã tham khảo các giáo trình và tài liệu giảng dạy môn học này của một số trường đại học trong và ngoài nước để giáo trình vừa đạt yêu cầu

cao về nội dung vừa thích hợp với đối tượng là học sinh - sinh viên của trường Cao đẳng Giao thông vận tải Trung ương I.

Mặc dù có rất nhiều cố gắng khi biên soạn, nhưng không thể tránh khỏi những sai sót. Rất mong nhận được sự góp ý của các thầy cô và đồng nghiệp để giáo trình hoàn thiện hơn.

Trân trọng cảm ơn!

Hà Nội, ngày 15 tháng 05 năm 2019

NHÓM TÁC GIẢ

Mục tiêu:

Học xong chương này học sinh sẽ vận dụng 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)
- Ngôn ngữ biểu diễn giải thuật
- Ngôn ngữ sơ đồ (lưu đồ), sử dụng lưu đồ để biểu diễn các giải thuật.

Nội dung chương:**1.1. Giải thuật (Algorithm)**

Là một dãy các thao tác xác định trên một đối tượng, sao cho sau khi thực hiện một số hữu hạn các bước thì đạt được mục tiêu. Theo R.A.Kowalski thì bản chất của thuật giải:

Thuật giải = Logic + Điều khiển

* **Logic:** Đây là phần khá quan trọng, nó trả lời câu hỏi "Thuật giải làm gì, giải quyết vấn đề gì?", những yếu tố trong bài toán có quan hệ với nhau như thế nào v.v...

Chẳng hạn: Để giải một bài toán tính diện tích hình cầu, mà bạn không còn nhớ công thức tính hình cầu thì bạn không thể viết chương trình cho máy để giải bài toán này được.

* **Điều khiển:** Thành phần này trả lời câu hỏi: giải thuật phải làm như thế nào?. Chính là cách thức tiến hành áp dụng thành phần logic để giải quyết vấn đề.

1.2. Chương trình (Program)

Là một tập hợp các mô tả, các phát biểu, nằm trong một hệ thống qui ước về ý nghĩa và thứ tự thực hiện, nhằm điều khiển máy tính làm việc. Theo Niklaus Wirth thì:

Chương trình = Thuật toán + Cấu trúc dữ liệu

Các thuật toán và chương trình đều có cấu trúc dựa trên 3 cấu trúc điều khiển cơ bản:

* **Tuần tự (Sequential):** Các bước thực hiện tuần tự một cách chính xác từ trên xuống, mỗi bước chỉ thực hiện đúng một lần.

* **Chọn lọc (Selection):** Chọn 1 trong 2 hay nhiều thao tác để thực hiện.

* **Lặp lại (Repetition):** Một hay nhiều bước được thực hiện lặp lại một số lần.

Để viết được chương trình chuẩn xác. Ta thực hiện theo các bước sau:

Tim, xây dựng thuật giải (trên giấy) → viết chương trình trên máy

→ dịch chương trình → chạy và thử chương trình

1.3. Ngôn ngữ lập trình (Programming language)

Ngôn ngữ lập trình là hệ thống các ký hiệu tuân theo các qui ước về ngữ pháp và ngữ nghĩa, dùng để xây dựng thành các chương trình cho máy tính.

Một chương trình được viết bằng một ngôn ngữ lập trình cụ thể (ví dụ Pascal, C, ...) gọi là chương trình nguồn, chương trình dịch làm nhiệm vụ dịch chương trình nguồn thành chương trình thực thi được trên máy tính.

1.4. Ngôn ngữ tự nhiên

Được thực hiện qua các bước:

Bước 1: Phân tích vấn đề và xác định các đặc điểm. (xác định I-P-O)

Bước 2: Lập ra giải pháp. (đưa ra thuật giải)

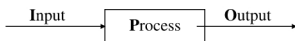
Bước 3: Cài đặt. (viết chương trình)

Bước 4: Chạy thử chương trình. (dịch chương trình)

Bước 5: Kiểm chứng và hoàn thiện chương trình. (thử nghiệm bằng nhiều số liệu và đánh giá)

I-P-O Cycle (Input-Process-Output Cycle) (Quy trình nhập-xử lý-xuất)

Ngôn ngữ tự nhiên sử dụng quy trình xử lý cơ bản của máy tính gồm I-P-O.



Ví dụ 1: Xác định Input, Process, Output của chương trình giải phương trình bậc nhất $ax + b = 0$

Input : hệ số a, b

Process : chia $-b$ cho a

Output : nghiệm x

Ví dụ 2: Xác định Input, Process, Output của chương trình tìm số lớn nhất của 2 số a và b .

Input : a, b

Process : Nếu $a > b$ thì

Output = a lớn nhất

Ngược lại *Output* = b lớn nhất

Ví dụ 3: Xác định Input, Process, Output của chương trình tính tiền lương công nhân tháng 10/2002 biết rằng lương = lương căn bản * ngày công

Input : lương căn bản, ngày công

Process : nhân lương căn bản với ngày công

Output : lương

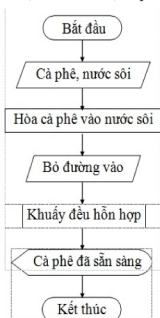
1.5. Ngôn ngữ sơ đồ (lưu đồ)

Sử dụng ngôn ngữ sơ đồ để dễ hơn về quy trình xử lý, các nhà lập trình đưa ra dạng lưu đồ để minh họa từng bước quá trình xử lý một vấn đề (bài toán).

Hình dạng (symbol)	Hành động (Activity)
--------------------	----------------------

Hình dạng (symbol)	Hành động (Activity)
	Dữ liệu vào (Input)
	Xử lý (Process)
	Dữ liệu ra (Output)
	Quyết định (Decision), sử dụng điều kiện
	Luồng xử lý (Flow lines)
	Gọi CT con, hàm... (Procedure, Function...)
	Bắt đầu, kết thúc (Begin, End)
	Điểm ghép nối (Connector)

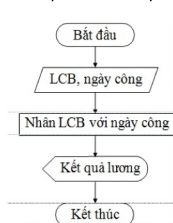
Ví dụ 1: Chuẩn bị cà phê



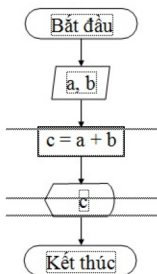
Ví dụ 2: Mô tả ví dụ 1



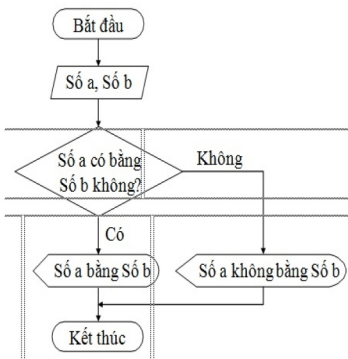
Ví dụ 3: Mô tả ví dụ 3



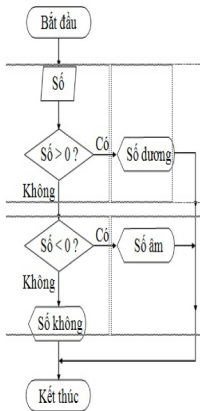
Ví dụ 4: Cộng hai số



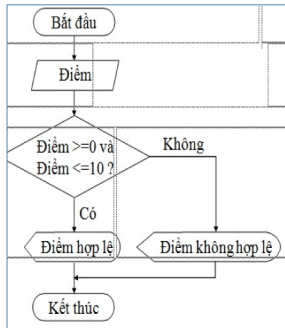
Ví dụ 5: So sánh 2 số



Ví dụ 7: Kiểm tra loại số



Ví dụ 8: Kiểm tra tính hợp lệ của điểm



Mục tiêu:

Học xong chương này, học sinh sẽ vận dụng các kiến thức kỹ năng cơ bản sau:

- Tổng quan về ngôn ngữ C
- Môi trường làm việc và cách sử dụng Turbo C 3.0
- Cách lập trình trên C
- Tiếp cận một số lệnh đơn giản thông qua các ví dụ
- Nắm bắt được một số kỹ năng đơn giản

Nội dung chương:

2.1 Tổng quan về ngôn ngữ lập trình 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.

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. 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.

Ngôn ngữ C có những đặc điểm cơ bản sau:

- *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.
- *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.
- *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 và cú pháp, có thể thu gọn kích thước của các mã lệnh làm chương trình chạy nhanh hơn.
- *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.

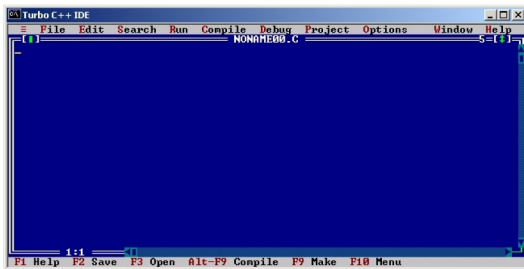
2.2 Môi trường làm việc và cách sử dụng Turbo C 3.0.

Turbo C là môi trường hỗ trợ lập trình C do hãng Borland cung cấp. Môi trường này cung cấp các chức năng như: soạn thảo chương trình, dịch, thực thi chương trình... Phiên bản được sử dụng ở đây là Turbo C 3.0.

2.2.1 Khởi động và thoát Borland C.

2.2.1.1. Khởi động

Gọi TurboC cũng giống như chạy các chương trình khác trong môi trường DOS hay Windows, màn hình sẽ xuất hiện menu của Turbo C có dạng như sau:



Dòng trên cùng gọi là thanh menu (menu bar). Mỗi mục trên thanh menu lại có thể có nhiều mục con nằm trong một menu kéo xuống.

Dòng dưới cùng ghi chức năng của một số phím đặc biệt. Chẳng hạn khi gõ phím F1 thì ta có được một hệ thống trợ giúp mà ta có thể tham khảo nhiều thông tin bổ ích.

Muốn vào thanh menu ngang ta gõ phím F10. Sau đó dùng các phím mũi tên qua trái hoặc phải để di chuyển vùng sáng tới mục cần chọn rồi gõ phím Enter. Trong menu kéo xuống ta lại dùng các phím mũi tên lên xuống để di chuyển vùng sáng tới mục cần chọn rồi gõ Enter.

Ta cũng có thể chọn một mục trên thanh menu bằng cách giữ phím Alt và gõ vào một ký tự đại diện của mục đó (ký tự có màu sắc khác với các ký tự khác). Chẳng hạn để chọn mục File ta gõ Alt-F (F là ký tự đại diện của File)

2.2.1.2. Thoát

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

Hoặc ấn tổ hợp phím **Alt - X**.

2.2.2 Dịch và Chạy chương trình.

Để thực hiện chương trình (Dịch và chạy chương trình, ta sử dụng Ctrl-F9 (giữ phím Ctrl và gõ phím F9).

Sau khi bạn nhập xong đoạn chương trình vào máy. Bạn Ấn và giữ phím Ctrl, gõ

F9 để dịch và chạy chương trình. Khi đó bạn 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, bạn ấn phím bất kỳ để quay về màn hình soạn thảo chương trình.

2.2.3. Các ví dụ

2.2.3.1. Ví dụ 1

Dòng	File Edit Search Run Compile Debug Project Option Window Help
1	/* Chương trình in ra câu bài học C đầu tiên */
2	#include <stdio.h>
3	
4	void main(void)
5	{
6	printf("Bài học C đầu tiên.");
7	}
	F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt - F9 Compile F9 Make F10 Menu

➤ Kết quả in ra màn hình

Bai học C đầu tiên. _

Dòng thứ 1: bắt đầu bằng /* và kết thúc bằng */ là hàng diễn giải (chú thích) giúp chương trình rõ ràng hơn. Khi dịch và chạy chương trình, dòng này không được dịch và cũng không thi hành lệnh gì cả.

Dòng thứ 2: Chứa phát biểu tiền xử lý #include <stdio.h>. Vì trong chương trình này ta sử dụng hàm thư viện của C là printf, do đó bạn 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 trống 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 thứ 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 học C đầu tiên"), 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, printf 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.

Bạn nhập đoạn chương trình trên vào máy. Dịch, chạy và quan sát kết quả.

Ctrl – F9: Dịch và chạy chương trình. **Alt – F5:** Xem màn hình kết quả.

➤ *Kết quả in ra màn hình*

Bai hoc C dau tien.

Ở dòng bạn vừa sửa có thêm \n, \n là ký hiệu xuống dòng sử dụng trong lệnh printf. Sau đây là một số ký hiệu khác.

+ *Các ký tự điều khiển:*

\n : Nhảy xuống dòng kế tiếp canh về cột đầu tiên.

\t : Canh cột tab ngang.

\r : Nhảy về đầu hàng, không xuống hàng.

\a : Tiếng kêu bip.

+ *Các ký tự đặc biệt:*

\\ : In ra dấu \

\" : In ra dấu "

\' : In ra dấu '

2.2.3.2. Ví dụ 2

Dòng	File Edit Search Run Compile Debug Project Option Window Help
1	/* Chương trình nhập và in ra màn hình giá trị biến*/
2	#include <stdio.h>
3	#include <conio.h>
4	
5	void main(void)
6	{
7	int i;
8	printf("Nhập vào một số: ");
9	scanf("%d", &i);
10	printf("Số bạn vừa nhập là: %d.\n", i);
11	getch();
12	}
	F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt - F9 Compile F9 Make F10 Menu

➤ *Kết quả in ra màn hình*

Nhập vào một số: 15
Số bạn vừa nhập là: 15.

Dòng thứ 7: *int i*; là lệnh khai báo, mẫu tự *i* gọi là tên biến. Biến là một vị trí trong bộ nhớ dùng lưu trữ giá trị nào đó mà chương trình sẽ lấy để sử dụng. Mỗi biến phải thuộc một kiểu dữ liệu. Trong trường hợp này ta sử dụng biến *i* kiểu số nguyên (integer) viết tắt là *int*.

Dòng thứ 9: *scanf("%d", &i)*. Sử dụng hàm *scanf* để nhận từ người sử dụng một trị nào đó. Hàm *scanf* trên có 2 đối mục. Đối mục *"%d"* được gọi là chuỗi định dạng, cho biết loại dữ kiện mà người sử dụng sẽ nhập vào. Chẳng hạn, ở đây phải nhập vào là số nguyên. Đối mục thứ 2 *&i* có dấu *&* đi đầu gọi là address operator, dấu *&* phối hợp với tên biến cho hàm *scanf* biến đem trị gõ từ bàn phím lưu vào biến *i*.

Dòng thứ 10: *printf("So ban vua nhap la: %d.\n", i)*. Hàm này có 2 đối mục. Đối mục thứ nhất là một chuỗi định dạng có chứa chuỗi văn bản *So ban vua nhap la:* và *%d* (ký hiệu khai báo chuyển đổi dạng thức) cho biết số nguyên sẽ được in ra. Đối mục thứ 2 là *i* cho biết giá trị lấy từ biến *i* để in ra màn hình.

Bạn nhập đoạn chương trình trên vào máy. Dịch, chạy và quan sát kết quả.

2.2.3.3. Ví dụ 3

Dòng	File Edit Search Run Compile Debug Project Option Window Help
1	/* Chương trình nhập vào 2 số
2	a, b in ra tổng*/ #include
3	<stdio.h>
4	#include <conio.h>
5	
6	void main(void)
7	{
8	int a, b;
9	printf("Nhập vào số a: "); scanf("%d", &a);
10	printf("Nhập vào số b: "); scanf("%d", &b);
11	printf("Tổng của 2 số %d và %d là %d.\n", a,
12	b, a+b); getch();
13	}
14	
	F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt - F9 Compile F9 Make

➤ **Kết quả in ra màn hình:**

Nhập vào số a: 4
Nhập vào số b: 14
Tổng của 2 số 4 và 14 là 18.
—