Trường Đại học Khoa học Tự nhiên
Khoa Công nghệ thông tin
Bộ môn Tin học cơ sở

NHẬP MÔN LẬP TRÌNH

Đặng Bình Phương
dbphuong@fit.hcmuns.edu.vn

CÁC KIỂU DỮ LIỆU CƠ SỞ

Nội dung

Các kiểu dữ liệu cơ sở

- Turbo C có 4 kiểu cơ sở như sau:
  - Kiểu số nguyên: giá trị của nó là các số nguyên như 2912, -1706, …
  - Kiểu số thực: giá trị của nó là các số thực như 3.1415, 29.12, -17.06, …
  - Kiểu luận lý: giá trị đúng hoặc sai.
  - Kiểu ký tự: 256 ký tự trong bảng mã ASCII.

Kiểu số nguyên

- Các kiểu số nguyên (có dấu)
  - n bit có dấu: -2^{n-1} … $+2^{n-1} - 1$

Kiểu (Type)	Độ lớn (Byte)	Miền giá trị (Range)
char	1	-128 … +127
int	2	-32.768 … +32.767
short	2	-32.768 … +32.767
long	4	-2.147.483.648 … +2.147.483.647

Kiểu số nguyên

- Các kiểu số nguyên (không dấu)
  - n bit không dấu: 0 … $2^n - 1$

Kiểu (Type)	Độ lớn (Byte)	Miền giá trị (Range)
unsigned char	1	0 … 255
unsigned int	2	0 … 65.535
unsigned short	2	0 … 65.535
unsigned long	4	0 … 4.294.967.295

Kiểu số thực

- Các kiểu số thực (floating-point)
  - Ví dụ
    - 17.06 = 1.706*10 = 1.706*10^1

Kiểu (Type)	Độ lớn (Byte)	Miền giá trị (Range)
float (*)	4	$3.4*10^{-38}$ … $3.4*10^{38}$
double (**)	8	$1.7*10^{-308}$ … $1.7*10^{308}$

- (*) Độ chính xác đơn (Single-precision) chính xác đến 7 số lẻ.
- (**) Độ chính xác kép (Double-precision) chính xác đến 19 số lẻ.

Kiểu luận lý

- Đặc điểm
  - C ngầm định một cách không tường minh:
    - false (sai): giá trị 0.
    - true (đúng): giá trị khác 0, thường là 1.
  - C++: bool
- Ví dụ
  - 0 (false), 1 (true), 2 (true), 2.5 (true)
  - 1 > 2 (0, false), 1 < 2 (1, true)

Kiểu ký tự

- Đặc điểm
  - Tên kiểu: char
  - Miền giá trị: 256 ký tự trong bảng mã ASCII.
  - Chính là kiểu số nguyên do:
    - Lưu tất cả dữ liệu ở dạng số.
    - Không lưu trực tiếp ký tự mà chỉ lưu mã ASCII của ký tự đó.
- Ví dụ
  - Lưu số 65 tương đương với ký tự 'A'…
  - Lưu số 97 tương đương với ký tự 'a'.

Biến

Biến

Ví dụ
```
int i;
int j, k;
unsigned char dem;
float ketqua, delta;
```

Cú pháp
```
<kiểu> <tên biến>;
<kiểu> <tên biến 1>, <tên biến 2>;
```


Hằng số

Hằng thường

Cú pháp
```
<kiểu> <tênhằng> = <giá trị>;
```

Ví dụ
```
int a = 1506;        // 1506_10
int b = 01506;       // 1506_8
int c = 0x1506;      // 1506_16 (0x hay 0X)
float d = 15.06e-3;  // 15.06*10^-3 (e hay E)
```


Hằng số

Hằng ký hiệu

Cú pháp
```
#define <tênhằng> <giá trị>
hoặc sử dụng từ khóa const.
```

Ví dụ
```
#define MAX 100          // Không có ;
#define PI 3.14          // Không có ;
const int MAX = 100;
const float PI = 3.14;
```


Biểu thức

- Khái niệm
  - Tạo thành từ các toán tử (Operator) và các toán hạng (Operand).
  - Toán tử tác động lên các giá trị của toán hạng và cho giá trị có kiểu nhất định.
  - Toán tử: +, –, *, /, %….
  - Toán hạng: hằng, biến, lời gọi hàm...
- Ví dụ
  - 2 + 3, a / 5, (a + b) * 5, …

Toán tử gán

- Khái niệm
  - Thường được sử dụng trong lập trình.
  - Gán giá trị cho biến.
- Cú pháp
  - <biến> = <giá trị>;
  - <biến> = <biến>;
  - <biến> = <biểu thức>;
  - Có thể thực hiện liên tiếp phép gán.

Toán tử gán

- Ví dụ

```
void main()
{
	int a, b, c, d, e, thuong;
	a = 10;
	b = a;
	thuong = a / b;
	a = b = c = d = e = 156;
	e = 156;
	d = e;
	c = d;
	b = c;
	a = b;
}
```


Các toán tử toán học

- Toán tử 1 ngôi
  - Chỉ có một toán hạng trong biểu thức.
  - ++ (tăng 1 đơn vị), -- (giảm 1 đơn vị)
  - Đặt trước toán hạng
    - Ví dụ ++x hay --x: thực hiện tăng/giảm trước.
  - Đặt sau toán hạng
    - Ví dụ x++ hay x--: thực hiện tăng/giảm sau.
- Ví dụ
  - x = 10; y = x++; // y = 10 và x = 11
  - x = 10; y = ++x; // x = 11 và y = 11

Các toán tử toán học

- Toán tử 2 ngôi
  - Có hai toán hạng trong biểu thức.
  - +, –, *, /, % (chia lấy phần dư)
  - x = x + y ⇔ x += y;
- Ví dụ
  - a = 1 + 2; b = 1 – 2; c = 1 * 2; d = 1 / 2;
  - e = 1*1.0 / 2; f = float(1) / 2; g = float(1 / 2);
  - h = 1 % 2;
  - x = x * (2 + 3*5); ⇔ x *= 2 + 3*5;

Các toán tử trên bit

- Các toán tử trên bit
  - Tác động lên các bit của toán hạng (nguyên).
  - & (and), | (or), ^ (xor), ~ (not hay lấy số bù 1)
  - >> (shift right), << (shift left)
  - Toán tử gộp: &=, |=, ^=, ~=, >>=, <<=

&	0	1
0	0	0
1	0	**1**

|	0	1
0	0	**1**
1	**1**	**1**

^	0	1
0	0	**1**
1	**1**	0

~	0	1
	1	0

Các toán tử trên bit

- Ví dụ

```
void main()
{
        int a = 5;  // 0000 0000 0000 0101
        int b = 6;  // 0000 0000 0000 0110

        int z1, z2, z3, z4, z5, z6;
        z1 = a & b; // 0000 0000 0000 0100
        z2 = a | b; // 0000 0000 0000 0111
        z3 = a ^ b; // 0000 0000 0000 0011
        z4 = ~a;    // 1111 1111 1111 1010
        z5 = a >> 2;// 0000 0000 0000 0001
        z6 = a << 2;// 0000 0000 0001 0100
}
```


Các toán tử quan hệ

- Các toán tử quan hệ
  - So sánh 2 biểu thức với nhau
  - Cho ra kết quả 0 (hay false nếu sai) hoặc 1 (hay true nếu đúng)
  - ==, >, <, >=, <, <=, !=
- Ví dụ
  - s1 = (1 == 2); s2 = (1 != 2);
  - s3 = (1 > 2); s4 = (1 >= 2);
  - s5 = (1 < 2); s6 = (1 <= 2);

Các toán tử luận lý

- Các toán tử luận lý
  - Tổ hợp nhiều biểu thức quan hệ với nhau.
  - && (and), || (or), ! (not)

&&	0	1
0	0	0
1	0	**1**

||	0	1
0	0	**1**
1	**1**	**1**

  - Ví dụ
    - s1 = (1 > 2) && (3 > 4);
    - s2 = (1 > 2) || (3 > 4);
    - s3 = !(1 > 2);

Toán tử điều kiện

- Toán tử điều kiện
  - Đây là toán tử 3 ngôi (gồm có 3 toán hạng)
  - <biểu thức 1> ? <biểu thức 2> : <biểu thức 3>
    - <biểu thức 1> đúng thì giá trị là <biểu thức 2>.
    - <biểu thức 1> sai thì giá trị là <biểu thức 3>.
- Ví dụ
  - s1 = (1 > 2) ? 2912 : 1706;
  - int s2 = 0;
  - 1 < 2 ? s2 = 2912 : s2 = 1706;

Toán tử phẩy

- Toán tử phẩy
  - Các biểu thức đặt cách nhau bằng dấu ,
  - Các biểu thức con lần lượt được tính từ trái sang phải.
  - Biểu thức mới nhận được là giá trị của biểu thức bên phải cùng.
- Ví dụ
  - x = (a++, b = b + 2);
  - ⇔ a++; b = b + 2; x = b;

Độ ưu tiên của các toán tử

Toán tử	Độ ưu tiên
() [] -> .	→
! ++ -- - + * (cast) & sizeof	←
* / %	→
+ -	→
<< >>	→
< <= > >=	→
== !=	→
&	→
|	→
^	→
&&	→
||	→
?:	←
= += -= *= /= %= &= ...	←
,	←

Độ ưu tiên của các toán tử

- Quy tắc thực hiện
  - Thực hiện biểu thức trong ( ) sâu nhất trước.
  - Thực hiện theo thứ tự ưu tiên các toán tử.

  => Tự chủ động thêm ( )
- Ví dụ
  - n = 2 + 3 * 5;

    => n = 2 + (3 * 5);
  - a > 1 && b < 2

    => (a > 1) && (b < 2)

Viết biểu thức cho các mệnh đề

- x lớn hơn hay bằng 3

  x >= 3
- a và b cùng dấu

  ((a>0) && (b>0)) || ((a<0) && (b<0))

  (a>0 && b>0) || (a<0 && b<0)
- p bằng q bằng r

  (p == q) && (q == r) hoặc (p == q && q == r)
- –5 < x < 5

  (x > –5) && (x < 5) hoặc (x > –5 && x < 5)

Câu lệnh

- Khái niệm
  - Là một chỉ thị trực tiếp, hoàn chỉnh nhằm ra lệnh cho máy tính thực hiện một số tác vụ nhất định nào đó.
  - Trình biên dịch bỏ qua các khoảng trắng (hay tab hoặc xuống dòng) chen giữa lệnh.
- Ví dụ

```
a=2912;
a = 2912;
a
=
2912;
```


Câu lệnh

- Phân loại
  - Câu lệnh đơn: chỉ gồm một câu lệnh.
  - Câu lệnh phức (khối lệnh): gồm nhiều câu lệnh đơn được bao bởi { và }
- Ví dụ

```
a = 2912;           // Câu lệnh đơn

{                   // Câu lệnh phức/khối lệnh
      a = 2912;
      b = 1706;
}
```


Câu lệnh xuất

- Thư viện
  - #include <stdio.h> (standard input/output)
- Cú pháp
  - printf(<chuỗi định dạng>[, <ds1>, <ds2>, …]);
  - <chuỗi định dạng> là cách trình bày thông tin xuất và được đặt trong cặp nháy kép “ ”.
    - Văn bản thường (literal text)
    - Ký tự điều khiển (escape sequence)
    - Đặc tả (conversion specifier)

Chuỗi định dạng

- Văn bản thường (literal text)
  - Được xuất y hệt như lúc gõ trong chuỗi định dạng.
- Ví dụ
  - Xuất chuỗi Hello World
    - ➔ printf(“Hello ”); printf(“World”);
    - ➔ printf(“Hello World”);
  - Xuất chuỗi a + b
    - ➔ printf(“a + b”);

Chuỗi định dạng

- Ký tự điều khiển (escape sequence)
  - Gồm dấu \ và một ký tự như trong bảng sau:

Ký tự điều khiển	Ý nghĩa
\a	Tiếng chuông
\b	Lùi lại một bước
\n	Xuống dòng
\t	Dấu tab
\\	In dấu \
\?	In dấu ?
\”	In dấu “

- Ví dụ
  - printf(“\t”); printf(“\n”);
  - printf(“\t\n”);

Chuỗi định dạng

- Đặc tả (conversion specifier)
  - Gồm dấu % và một ký tự.
  - Xác định kiểu của biến/giá trị muốn xuất.
  - Các đối số chính là các biến/giá trị muốn xuất, được liệt kê theo thứ tự cách nhau dấu phẩy.

Đặc tả	Ý nghĩa	
%c	Ký tự	char
%d, %ld	Số nguyên có dấu	int, short, long
%f, %lf	Số thực	float, double
%s	Chuỗi ký tự	char[], char*
%u	Số nguyên không dấu	unsigned int/short/long

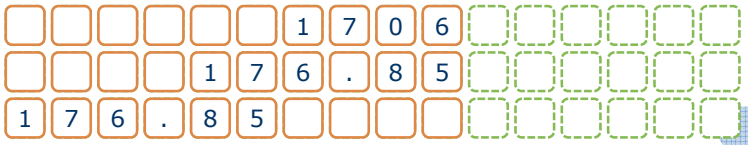
Chuỗi định dạng

- Ví dụ
  - int a = 10, b = 20;
  - printf(“%d”, a); ➔ Xuất ra 10
  - printf(“%d”, b); ➔ Xuất ra 20
  - printf(“%d %d”, a, b); ➔ Xuất ra 10 20
  - float x = 15.06;
  - printf(“%f”, x); ➔ Xuất ra 15.060000
  - printf(“%f”, 1.0/3); ➔ Xuất ra 0.333333

Định dạng xuất

- Cú pháp
  - Định dạng xuất số nguyên: %nd
  - Định dạng xuất số thực: %n.kd

```
int a = 1706;
float x = 176.85;
printf("%10d", a);printf("\n");
printf("%10.2f", x);printf("\n");
printf("%.2f", x);printf("\n");
```



Chuỗi định dạng

- Phối hợp các thành phần
  - int a = 1, b = 2;
  - Xuất 1 cong 2 bang 3 và xuống dòng.
    - printf("%d", a); // Xuất giá trị của biến a
    - printf(" cong "); // Xuất chuỗi " cong "
    - printf("%d", b); // Xuất giá trị của biến b
    - printf(" bang "); // Xuất chuỗi " bang "
    - printf("%d", a + b); // Xuất giá trị của a + b
    - printf("\n"); // Xuất điều khiển xuống dòng \n
  - ➔ printf("%d cong %d bang %d\n", a, b, a+b);

Câu lệnh nhập

- Thư viện
  - #include <stdio.h> (standard input/output)
- Cú pháp
  - scanf(<chuỗi định dạng>[, <đs1>, <đs1>, …]);
  - <chuỗi định dạng> giống định dạng xuất nhưng chỉ có các đặc tả.
  - Các đối số là tên các biến sẽ chứa giá trị nhập và được đặt trước dấu &

Câu lệnh nhập

- Ví dụ, cho a và b kiểu số nguyên
  - scanf("%d", &a); // Nhập giá trị cho biến a
  - scanf("%d", &b); // Nhập giá trị cho biến b
  - ➔ scanf("%d%d", &a, &b);
  - Các câu lệnh sau đây sai
    - scanf("%d", a); // Thiếu dấu &
    - scanf("%d", &a, &b);// Thiếu %d cho biến b
    - scanf("%f", &a); // a là biến kiểu số nguyên
    - scanf("%9d", &a); // không được định dạng
    - scanf("a = %d, b = %d", &a, &b");

Một số hàm hữu ích khác

- Các hàm trong thư việc toán học
  - #include <math.h>
  - 1 đầu vào: double, Trả kết quả: double
    - acos, asin, atan, cos, sin, …
    - exp, log, log10
    - sqrt
    - ceil, floor
    - abs, fabs
  - 2 đầu vào: double, Trả kết quả: double
    - double pow(double x, double y)

Một số hàm hữu ích khác

- Ví dụ
  - int x = 4, y = 3, z = -5;
  - float t = -1.2;
  - float kq1 = sqrt(x1);
  - int kq2 = pow(x, y);
  - float kq3 = pow(x, 1/3);
  - float kq4 = pow(x, 1.0/3);
  - int kq5 = abs(z);
  - float kq6 = fabs(t);

Bài tập lý thuyết

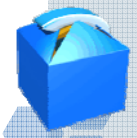
1. Trình bày các kiểu dữ liệu cơ sở trong C và cho ví dụ.
2. Trình bày khái niệm về biến và cách sử dụng lệnh gán.
3. Phân biệt hằng thường và hằng ký hiệu. Cho ví dụ minh họa.
4. Trình bày khái niệm về biểu thức.
   Tại sao nên sử dụng cặp ngoặc đơn.
5. Trình bày cách định dạng xuất.

Bài tập thực hành

4. Nhập năm sinh của một người và tính tuổi của người đó.
5. Nhập 2 số a và b. Tính tổng, hiệu, tính và thương của hai số đó.
6. Nhập tên sản phẩm, số lượng và đơn giá. Tính tiền và thuế giá trị gia tăng phải trả, biết:
   a. tiền = số lượng * đơn giá
   b. thuế giá trị gia tăng = 10% tiền

Bài tập thực hành

7. Nhập điểm thi và hệ số 3 môn Toán, Lý, Hóa của một sinh viên. Tính điểm trung bình của sinh viên đó.
8. Nhập bán kính của đường tròn. Tính chu vi và diện tích của hình tròn đó.
9. Nhập vào số xe (gồm 4 chữ số) của bạn. Cho biết số xe của bạn được mấy nút?



Bài tập 4

```
#include <stdio.h>
#include <conio.h>

void main()
{
	int NamSinh, Tuoi;
	printf("Nhap nam sinh: ");
	scanf("%d", &NamSinh);
	Tuoi = 2007 - NamSinh;
	printf("Tuoi cua ban la %d", Tuoi);
	getch();
}
```


Bài tập 5

```
#include <stdio.h>
#include <conio.h>

void main()
{
	int a, b;
	printf("Nhap hai so nguyen: ");
	scanf("%d%d", &a, &b);
	Tong = a + b; Hieu = a - b;
	Tich = a * b; Thuong = a / b;
	printf("Tong cua a va b: %d", Tong);
	printf("Hieu cua a va b: %d", Hieu);
	printf("Tich cua a va b: %d", Tich);
	printf("Thuong cua a va b: %d", Thuong);
}
```


Bài tập 6

```
#include <stdio.h>
#include <conio.h>

void main()
{
	int SoLuong, DonGia, Tien;
	float VAT;

	printf("Nhap so luong va don gia: ");
	scanf("%d%d", &SoLuong, &DonGia);
	Tien = SoLuong * DonGia;
	VAT = Tien * 0.1;
	printf("Tien phai tra: %d", Tien);
	printf("Thue phai tra: %.2f", VAT);
}
```

Bài tập 7

```
#include <stdio.h>
#include <conio.h>

void main()
{
	float T, L, H, DTB;
	int HsT, HsL, HsH;
	printf("Nhap diem Toan, Ly, Hoa: ");
	scanf("%f%f%f", &T, &L, &H);
	printf("Nhap he so Toan, Ly, Hoa: ");
	scanf("%d%d%d", &HsT, &HsL, &HsH);
	DTB = (T * HsT + L * HsL + H * HsH) /
		(HsT + HsL + HsH);
	printf("DTB cua ban la: %.2f", DTB);
}
```


Bài tập 8

```
#include <stdio.h>
#include <conio.h>
#define PI 3.14

void main()
{
	float R, ChuVi, DienTich;
	printf("Nhap ban kinh duong tron: ");
	scanf("%f", &R);
	ChuVi = 2*PI*R;
	DienTich = PI*R*R;
	printf("Chu vi: %.2f", ChuVi);
	printf("Dien tich: %.2f", DienTich);
}
```


Bài tập 9

```
#include <stdio.h>
#include <conio.h>

void main()
{
	int n;
	int n1, n2, n3, n4, SoNut;
	printf("Nhap bien so xe (4 so): ");
	scanf("%d", &n);
	n4 = n % 10; n = n / 10;
	n3 = n % 10; n = n / 10;
	n2 = n % 10; n = n / 10;
	n1 = n;
	SoNut = (n1 + n2 + n3 + n4) % 10;
	printf("So nut la: %d", SoNut);
}
```