

Bài 3: Kiểu dữ liệu chuẩn, Các hàm nhập/xuất.

1. Các kiểu dữ liệu trong C/C++

a. Kiểu số nguyên

Kiểu số nguyên bao gồm:

Tên kiểu	Số byte	Miền giá trị	Ghi chú
short	2	-32768 → 32767	
int	2	-32768 → 32767	
long	4	$2^{31} \rightarrow 2^{31}-1$	
long long	8	$2^{63} \rightarrow 2^{63}-1$	C++11

Các phép toán áp dụng được trên kiểu số nguyên:

- Các phép toán số học: +, -, *, /, % (modulo)
- Các phép toán so sánh: <, <=, >, >=, ==, !=
- Các phép toán dịch chuyển số học: >>, và <<
- Các phép toán trên bit: ~ (not bit), & (and bit), | (or bit), ^ (xor bit)

b. Kiểu số thực

Kiểu số thực bao gồm:

Tên kiểu	Số byte	Miền giá trị	Ghi chú
float	4	3.4E-38 → 3.4E+38	Độ chính xác đến 8 chữ số
double	8	1.7E-308 → 1.7E+30	Độ chính xác đến 16 chữ số
long double	10	3.4E-4932 → 1.1E4932	Độ chính xác đến 18 chữ số

Các phép toán trên số thực

- Các phép toán số học: +, -, *, / (phép % không thực hiện được trên số thực)
- Các phép toán so sánh: <, <=, >, >=, ==, !=

c. Kiểu ký tự

Kiểu ký tự bao gồm:

Tên kiểu	Số byte	Miền giá trị	Ghi chú
char (signed char)	2	-128 → 127	
unsigned char	2	0 → 255	

Các phép toán thực hiện được trên số nguyên đều thực hiện được trên kiểu ký tự vì bản chất của kiểu ký tự là kiểu số nguyên. Giá trị của nó là mã của ký tự.

Ví dụ:

Ký tự	0	1	A	B	C	a	b	c	
Mã	48	49	65	66	67	97	98	99	

Để mở rộng các kiểu dữ liệu cơ sở, C/C++ đưa thêm các tiền tố (prefix): short, long, unsigned, signed vào trước tên các kiểu cơ sở.

Ví dụ:

unsigned int	2 bytes	(0 .. 65535)
short int	2 bytes	(-32768 .. 32767)
long int	4 bytes	($2^{31} \rightarrow 2^{31}-1$)

d. Kiểu logic

Tên kiểu: bool

Số byte: 1.

Miền giá trị: true ($\neq 0$), false ($= 0$)

Các phép toán:

- Phép and ký hiệu: &&
- Phép or ký hiệu: ||
- Phép not ký hiệu: !

(§) Ta có thể sử dụng `typedef` kiểu `tên_mới`; để định nghĩa lại tên kiểu dữ liệu.

Ví dụ: `typedef float sothuc;`

2. Các hàm nhập xuất trong C/C++

a. Nhập xuất trong C

Để sử dụng các hàm nhập xuất dữ liệu trong C, cần khai báo thư viện: `#include <stdio.h>`

Hàm xuất dữ liệu: `printf(<khung định dạng>, [<ds tham số>]);`

Trong đó:

- `<ds tham số>`: có thể là hằng, biến, biểu thức, hàm...

- <khung định dạng> được đặt trong cặp dấu “” bao gồm: Ký tự thường, ký tự điều khiển hoặc mã định dạng các tham số và cách biểu thị tham số trên màn hình. Mỗi mã định dạng bắt đầu bằng dấu %. Số lượng mã định dạng bằng số tham số.

Dạng tổng quát của mã định dạng: %[flag][width].[prec]<mã định dạng>

**flag* là + (có dấu), -(căn trái), khoảng trắng hoặc không có: dùng để căn dữ liệu. Khoảng trắng: nếu dữ liệu âm có dấu -, dương thì dấu + thay bằng khoảng trắng.

**width*: là độ rộng dùng để in dữ liệu

**prec*: là số chữ số có nghĩa sau dấu chấm thập phân (đối với số thực).

Mã định dạng được quy định trong bảng sau:

Mã	Tham số	Dạng xuất
d	Số nguyên	Số nguyên int
u	Số nguyên	Số nguyên không dấu
ld	Số nguyên	long
lu	Số nguyên	Unsigned long
o	Số nguyên	Số nguyên hệ 8
x,X	Số nguyên	Số nguyên hệ 16
f	Số thực	Số thực float, double (dấu chấm tĩnh)
e,E	Số thực	Số thực float, double (biểu diễn dạng khoa học) (e)
g,G	Số thực	Biểu diễn khoa học (e)
c	Ký tự	char
s	Xâu ký tự	Xâu ký tự
p	Con trỏ	Địa chỉ

Hàm nhập dữ liệu: **scanf(<khung định dạng>,<danh sách tham số>);**

Trong đó:

- <khung định dạng>: tương tự hàm printf

- <danh sách tham số>: là dãy địa chỉ các biến tương ứng với mã định dạng và giữa chúng được ngăn cách nhau bởi dấu phẩy. Để lấy địa chỉ biến dùng ký hiệu & đặt trước tên biến.

a. Nhập xuất trong C++

Để nhập xuất dữ liệu trong C++, cần khai báo thư viện: #include <iostream>

Lệnh nhập dữ liệu: cin >> bt1 >> bt2 ...;

Lệnh xuất dữ liệu: cout << bt1 << bt2 ...;

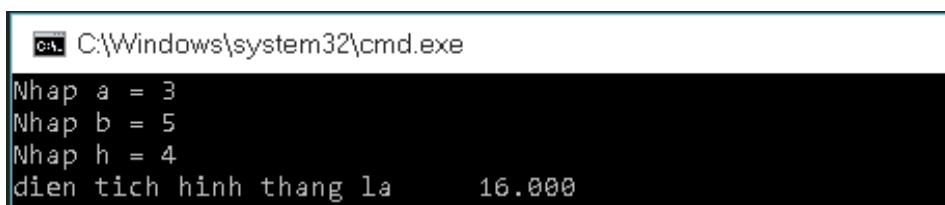
Trong C++ tự động kiểm tra kiểu dữ liệu nên đối với lệnh nhập/ xuất dữ liệu không sử dụng khung định dạng như trong C.

3. Một số ví dụ

Ví dụ 1: Viết chương trình tính diện tích hình thang có độ dài hai đáy là a và b, chiều cao là h. Các giá trị a, b và h là số nguyên.

```
#include <stdio.h>
int a,b,h;
float s;
int main(){
    printf("Nhập a = ");
    scanf("%d",&a);
    printf("Nhập b = ");
    scanf("%d",&b);
    printf("Nhập h = ");
    scanf("%d",&h);
    s=(a+b)*h/2;
    printf("diện tích hình thang là %10.3f",s);
    return 0;
}
```

Kết quả chạy chương trình



```
C:\Windows\system32\cmd.exe
Nhập a = 3
Nhập b = 5
Nhập h = 4
diện tích hình thang là      16.000
```

Giải thích mã định dạng %10.3f nghĩa là dùng 10 khoảng để in giá trị biến s (kiểu float) và có 3 chữ số thập phân.

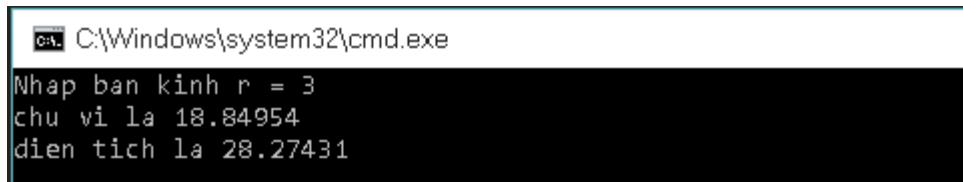
Cũng với bài toán trên, nếu sử dụng lệnh nhập xuất của C++ thì ta có thể thay hàm printf và scanf bằng cout và cin như sau:

```
#include <iostream>
using namespace std;
int a,b,h;
float s;
int main(){
    cout << "Nhập a = ";
    cin >> a;
    cout << "Nhập b = ";
    cin >> b;
    cout << "Nhập h = ";
    cin >> h;
    s=(a+b)*h/2;
    cout << "diện tích hình thang là: " << s;
    return 0;
}
```

Ví dụ 2: Viết chương trình tính chu vi và diện tích hình tròn với bán kính r được nhập từ bàn phím.

```
#include <stdio.h>
#include <math.h>
const float pi = 3.14159;
float c,s,r;
int main(){
    printf("Nhập bán kính r = ");
    scanf("%f",&r);
    c=2*pi*r;
    s=pow(r,2)*pi;
    printf("chu vi là %0.5f \n",c);
    printf("diện tích là %0.5f \n",s);
    return 0;
}
```

Kết quả chạy chương trình



```
C:\Windows\system32\cmd.exe
Nhập bán kính r = 3
chu vi là 18.84954
diện tích là 28.27431
```

Giải thích:

- + Hàm pow (x, n) là hàm tính x^n được cung cấp trong thư viện <math.h>
- + Khung định dạng %0.5f nghĩa là in số thực (float) với số khoảng (0) do chương trình tự động điều chỉnh và có 5 chữ số thập phân.
- + Ký tự điều khiển \n là ký tự chuyển con trỏ xuống dòng sau khi in.

Ví dụ 3: Viết chương trình nhập một ký tự và in mã ký tự ra màn hình

```
#include <stdio.h>
unsigned char ch;
int main(){
    printf("Nhập một ký tự: ");
    scanf("%c",&ch);
    printf("Giá trị ma của %c là %d",ch,ch);
    return 0;
}
```

Kết quả chạy chương trình



Giải thích: Có 2 ký tự trong khung định dạng là %c và %d tương ứng với hai giá trị in ra là ch và ch. Giá trị biến ch đầu khi in ra với định dạng %c nên chương trình sẽ in kiểu ký tự, Giá trị biến ch thứ 2 khi in ra với định dạng %d (số nguyên kiểu int) nên chương trình sẽ in kiểu số nguyên (chính mà mã của ký tự).