



Cấu trúc chương trình C

Các hàm nhập | xuất thông dụng



Nội dung chính

1

Cấu trúc chương trình

2

Hàm nhập / xuất

3

Bài tập



1. Cấu trúc chương trình

❖ Một chương trình (source code) C *chuẩn* gồm có các thành phần sau:

0. Chú thích chính cho toàn bộ chương trình
1. Các chỉ thị tiền xử lý
2. Khai báo các kiểu dữ liệu mới, nguyên mẫu hàm phụ (nếu có)
3. Khai báo hằng, khai báo biến; hàm phụ (nếu có)
4. Hàm chính (chương trình chính)

Ví dụ một chương trình C

```
/* Vi du minh hoa cua mot chuong trinh C */  
#include<stdio.h>  
void test();  
int x = 8;  
void test()  
{  
    printf("Day la lenh thuc hien boi ham phu: \n");  
    printf("Gia tri cua bien toan cuc x la: %d\n", x);  
}  
int main()  
{  
    printf("Gia tri cua bien toan cuc x la: %d\n", x);  
    test(); //goi ham phu test  
    return 1;  
}
```

← Đây là chú thích

← Đây là chỉ thị tiền xử lý

← Khai báo nguyên mẫu hàm phụ *test*

← Khai báo biến toàn cục *x*

← hàm phụ *test*

← Hàm chính



1.1. Chỉ thị tiền xử lý

- ❖ Vai trò của chỉ thị tiền xử lý.
- ❖ Một số chỉ thị tiền xử lý thông dụng:
 - Chỉ thị chèn tệp `# include`
 - Chỉ thị thay thế `# define`

Ghi chú: Kết thúc của hai dòng này KHÔNG bao giờ có dấu chấm phẩy ;



Chỉ thị tiền xử lý (tiếp)

❖ Chỉ thị chèn tệp:

include

Dùng để chèn *thư viện* hoặc *file* vào chương trình. *Thư viện* hoặc *file* đó chứa thông tin về các hàm hoặc các biến, hoặc các mảng ngoài được sử dụng trong chương trình

vd: Chèn thư viện chứa các hàm vào/ra trên thiết bị chuẩn:

#include <**stdio.h**>

❖ Chỉ thị thay thế:

define

- Dùng để thay thế

vd: **#define** **PI** 3.1415926

- Định nghĩa các macro có đối tượng tự như các hàm

vd: **#define** **max**(a, b) ((a) > (b)) ? (a) : (b)

1.2. Hàm chính

- Là phần bắt buộc của một chương trình C. Có dạng tổng quát như sau:

```
<kiểu của kết quả trả cho hàm> main( )  
{  
    Các khai báo biến;  
    Các lệnh thực thi;  
    return (Kết quả trả cho hàm);  
}
```

Opt. →

```
int main()  
{  
    int x;  
    scanf("%d", &x);  
    return 1;  
}
```

```
main()  
{  
    int x;  
    scanf("%d", &x);  
}
```

Ví dụ một chương trình đơn giản trong C

```
# include <stdio.h>  
# define PI 3.1415926
```

Chỉ thị tiền xử lý

```
main()
```

Tên hàm

```
{  
    printf("Pi value: %f\n", PI);  
}
```

Chương trình chính

Thân hàm



Một số lưu ý trong chương trình C

- ❖ Kết thúc câu lệnh phải có dấu chấm phẩy(;)
- ❖ Ngay sau tên hàm không có dấu chấm phẩy
- ❖ Cách chú thích một đoạn

*/**

Ghi chú đặt ở đây

Hoặc đoạn chương trình không thực thi

**/*

- ❖ Một câu được chú thích đặt phía sau cặp dấu: *//*
- ❖ Thân hàm phải được bao bởi cặp dấu: *{ }*



Nội dung chính

2

Các hàm nhập xuất dữ liệu



Hàm xuất dữ liệu | *printf*

❖ Hàm xuất thông tin ra stdout(màn hình): **printf**


■ Cú pháp tổng quát:

printf("chuỗi_in_ra_màn_hình", *biến1*, *biến2*,...);

■ chuỗi_in_ra_màn_hình có thể là:

- Chuỗi kí tự thông thường được in ra màn hình
- Các kí tự **đặc tả** cho các *biến* cần xuất dữ liệu ra màn hình
- Các kí tự điều khiển dòng

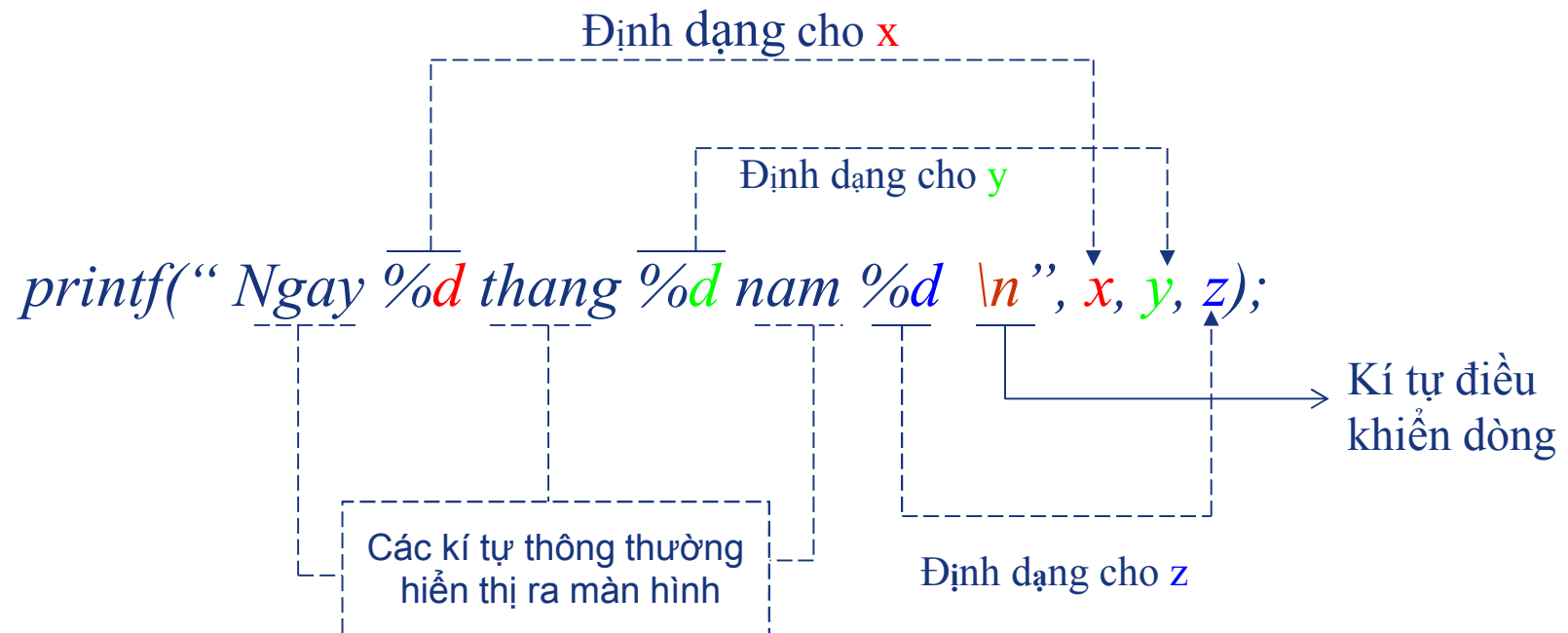
➤ Số kí tự **đặc tả** bằng số *biến*

- 
- ✓ Một số kí tự dùng để *đặc tả*(*định dạng*) cho dữ liệu chứa trong biến và dùng để *điều khiển dòng* thường sử dụng trong hàm **printf**

Đặc tả dữ liệu chứa trong biến		Kí tự điều khiển dòng	
Đặc tả	Kiểu dữ liệu	Mô tả	Mục đích
%c	Kí tự(character)	\n	Xuống dòng
%s	Chuỗi kí tự(string)	\t	Cách vào một tab
%d	Số nguyên(int)	\f	Sang trang mới
%f	Số thực(float)	\b	Lùi lại một kí tự
%lf	Số thực(double)	...	

Ví dụ về hàm printf

int $x = 2, y = 9, z = 1945;$





Ví dụ: Chương trình minh họa hàm **printf**

- ❖ Gõ lại đoạn chương trình sau, biên dịch(compile) và chạy(run). Nhận xét các thông tin hiện thị trên màn hình.

```
#include<stdio.h>
main()
{
    int  x = 1945;
    printf("\n1.Gia tri cua x la: ");
    printf("\n2.Gia tri cua x la: ", x);
    printf("\n3.Gia tri cua x la: %d ", x);
    printf("\n4.Gia tri cua x la: %d ");
    printf("\n5.Gia tri cua x la: %f ", x);
    printf("\nKet thuc chuong trinh");
}
```



Hàm xuất dữ liệu | *puts*

❖ Đưa một chuỗi kí tự ra stdout (màn hình): **puts**

- Cú pháp: `int puts(char *s)`

❖ Ví dụ

- `puts("Xin chào !");`

- `char x[] = "Eureka !";`
`puts(x);`



Hàm xuất dữ liệu | *putchar*

- ❖ Đẩy ra stdout(màn hình) một kí tự tương ứng với đầu vào là một số nguyên trong bảng mã ASCII

- Cú pháp `int putchar(int char);`

Kết quả trả về cho hàm này chính là số nguyên tương ứng với ký tự được đẩy ra stdout.

- Ví dụ:

```
putchar(65);           // A
```

```
putchar(97);           // a
```

```
putchar(90); // Z
```




Hàm nhập dữ liệu | *scanf*

❖ Nhận dữ liệu có đặc tả từ bàn phím: **scanf**

- *scanf* nhận từ stdin (bàn phím) dữ liệu có **đặc tả**(định dạng) và lưu giá trị vào **biến**
- Cú pháp: **scanf**("đặc_tả", &*biến1*, &*biến2*,...);
- Số **đặc_tả** phải bằng số **biến**. Các đặc tả viết liền nhau
- Tùy vào kiểu dữ liệu của biến để chọn đặc tả phù hợp. Một số đặc tả thông:

Kiểu dữ liệu	Đặc tả	Kiểu dữ liệu	Đặc tả
Kí tự(char)	%c	Số nguyên(int)	%d
Chuỗi kí tự	%s	Số thực(float)	%f
		Số thực(double)	%lf



Ví dụ: Dùng hàm **scanf** nhập dữ liệu từ bàn phím

```
int z;                                //khai báo biến  
scanf("%d", &z);                     //Nhập dữ liệu cho biến  
  
int x; float y;                       //khai báo biến  
scanf("%d%f ", &x, &y);              //nhập dữ liệu cho biến  
  
int d, m, y;                          //khai báo biến  
scanf("%d%d%d", &d, &m, &y);         //Nhập dữ liệu
```



Hàm nhập dữ liệu | *gets*

❖ Nhận từ stdin (bàn phím) một xâu kí tự: *gets*

- *gets* nhận từ bàn phím một xâu kí tự và lưu giá trị vào biến.
- Cú pháp `char *gets(char *s);`

➤ Ví dụ: Nhận vào từ bàn phím chuỗi kí tự.

```
char x[30]; //khai báo biến mảng x kiểu char để lưu xâu
```

```
gets(x); //nhận từ bàn phím một chuỗi và lưu vào biến x
```



Hàm nhập dữ liệu | *getchar*

❖ Nhận từ stdin (bàn phím) một kí tự: ***getchar***

- *getchar* nhận từ stdin (bàn phím) một kí tự và trả về là một số nguyên.
- Cú pháp: **int** ***getchar***();

❖ Ví dụ: Nhận từ stdin(bàn phím) một kí tự và trả về là một số nguyên trong bảng mã ASCII

- *int* *y*;
- *y* = ***getchar***();



Một số ví dụ: Tính tổng hai số bất kì

```
#include <stdio.h>  
main ()  
{  
    float a, b, s;  
  
    printf("Nhap vao so a: ");  
    scanf("%f", &a);  
    printf("Nhap vao so b: ");  
    scanf("%f", &b);  
    s=a+b;  
    printf("Tong cua 2 so la: %f \n",s);  
}
```



Tính chu vi, diện tích của một hình tròn

```
#include <stdio.h>
#include <math.h>
main()
{
    float  R, cv, dt;
    puts("Nhap vao ban kinh cua hinh tron ");
    scanf("%f",&R);
    cv = 2*R*M_PI;
    dt = M_PI*R*R;
    printf("Chu vi = %10.2f\nDien tich = %10.2f ",cv,dt);
}
```



Tóm tắt bài học

- ❖ Cấu trúc một chương trình C chuẩn
- ❖ Các hàm xuất dữ liệu
 - *printf*, *puts*, *putchar*
- ❖ Các hàm nhập dữ liệu
 - *scanf*, *gets*, *getchar*
- ❖ Một số ví dụ minh họa



Nội dung chính

3

Bài tập



Bài tập về nhà

1. Viết chương trình nhập vào bán kính đường tròn và tính chu vi, diện tích.
2. Viết chương trình nhập vào đáy, đường cao của tam giác và tính diện tích
3. Viết chương trình nhập vào cạnh của hình vuông và tính diện tích.
4. Viết chương trình nhập vào đáy lớn, đáy nhỏ của hình thang và tính diện tích.
5. Viết chương trình nhập vào một số a bất kỳ và in ra giá trị bình phương a , lập phương a và giá trị a^4 .