

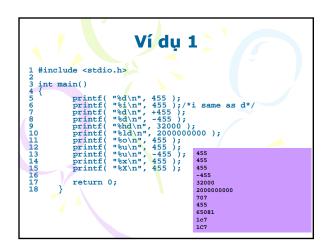
#### Vào/ra trong C

- Thư viện các hàm vào/ra chuẩn trong
   C được lưu trong file <stdio.h>.
- Streams
  - Chuỗi ký tư được tổ chức thành dòng
    - Kết thúc với ký tư "new line"
  - Có thể được chuyển hướng
    - Đầu vào chuẩn bàn phím
    - Đầu ra chuẩn màn hình
    - Lỗi chuẩn màn hinh

## • printf • printf • Dịnh dạng đầu ra với printf • Định dạng chính các đầu ra • Chỉ định sự chuyển đổi: cờ, độ rộng trường, độ chính xác, ... - Có thể làm tròn, dóng hàng cột, căn lề trái/phải, ... • Định dạng • printf( format-control-string, other-arguments ); - format control string: gồm 1 d/sách các kiểu dữ liệu dầu ra, các xâu và ký tự điều khiển - other-arguments: ứng với các kiểu dữ liệu chỉ định

trong format-control-string

#### Hiển thị các số nguyên Mô tả Định dạng %d Hiển thị số nguyên có dấu %i Hiển thị số nguyên có dấu. (i và d khi dùng với scanf sẽ có tác dụng khác nhau.) **%o** Hiển thị số bát phân không dấu %u Hiển thị số nguyên không dấu Hiển thị số nguyên không dấu hệ 16. X sử dụng %x or %X bộ (0-9, các ký tự A-F), x sử dụng bộ (0-9, các ký tự a-f) %h or %l Đặt trước các định dạng của số nguyên để chỉ (letter I) định số nguyên ngắn (h) hoặc dài (l).



# Hiển thị số thực dấu phẩy động - Dùng dấu chấm để phân cách (33.5) - Biểu diễn dưới dạng mũ • 150.3 = 1.503 x 10² = 1.503E+02 - f - in số thực với ít nhất 1 ký tự bên trái dấu phẩy - g (hay G) - in f hoặc e(E) bỏ qua số 0 thừa (1.2300 thành 1.23)

```
Ví du 2
#include <stdio.h>
int main()
        printf( "%e\n", 1234567.89 );
printf( "%e\n", +1234567.89 );
printf( "%e\n", -1234567.89 );
printf( "%e\n", 1234567.89 );
printf( "%f\n", 1234567.89 );
printf( "%g\n", 1234567.89 );
printf( "%g\n", 1234567.89 );
         return 0;
                                                                                           1.234568e+006
1.234568e+006
-1.234568e+006
1.234568E+006
1234567.890000
                                                                                            1.23457e+006
                                                                                            1.23457E+006
```

```
In xâu và ký tự
- In giá trị char
- Không dùng để in ký tự đầu tiên của xâu
- Cần 1 con trỏ trỏ đến biến char
- In các ký tự đến khi gặp ký tự NULL ('\0')
- Không dùng để in giá trị char
- Dùng dấu nháy đơn cho ký tự ('z')
- Dùng dấu nháy kép cho xâu ("z") (thực tế
 gồm 2 ký tư, 'z' và '\0')
```

```
Ví dụ 3
1 #include <stdio.h>
     int main()
{
           char character = 'A';
char string[] = "This is a string";
const char *stringPtr = "This is also a string";
           printf( "%c\n", character );
printf( "%s\n", "This is a string" );
printf( "%s\n", string );
printf( "%s\n", stringPtr );
12 printf( "7
13
14 return 0;
15 }
                                                                        A
This is a string
This is a string
This is also a string
```

#### Ví dụ 4 1#include <stdio.h> 3int main() 4 { int \*ptr; int x = 12345, y; ptr = &x; printf( "The value of ptr is %p\n", ptr ); printf( "The address of x is %p\n\n", &x ); y = printf( "This line has 28 characters\n" ); printf( "%d characters were printed\n\n", y ); printf( "Printing a %% in a format control string\n" ); The value of ptr is 0065FDF0 The address of x is 0065FDF0 return 0; Total characters printed on this line is: 41 Printing a % in a format control string

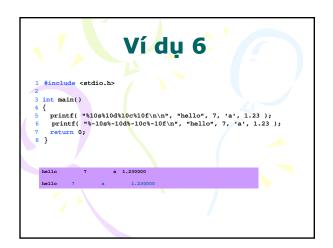
#### Các dạng chuyển đổi khác

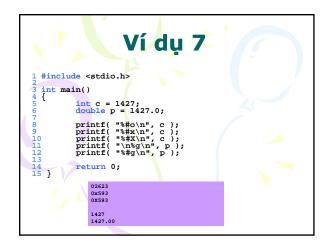
- - Hiển thị giá trị con trỏ (điạ chỉ)
- - Lưu số ký tư đã được in ra bởi câu lênh
  - Nhân tham số là con trỏ trỏ đến một số nguyên
  - Không hiển thi sử dụng %n
- In dấu %

### In với độ rộng trường và độ chính xác

- Đô rông trường
  - Là kích thước của trường dữ liệu được in ra
- - Đô chính xác: dùng (.), sau đó đến số con số sau %
  - %.3f
  - Có thể kết hợp với độ rộng trường %5.3f
  - Có thể dùng số nguyên để xác định độ rộng trường và độ chính xác
    - Dùng \*
    - printf( "%\*.\*f", 7, 2, 98.736 );







```
In các ký tự đặc biệt

Cú Mô tả
pháp

l' In dấu (')

l" In dấu (")

l? In dấu ()

la Tạo ra tiếng chuông hoặc thông báo dạng hình ảnh

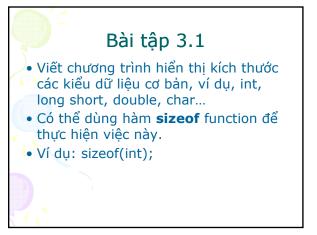
lb Chuyển con trở quay lui 1 vị trí ở dòng hiện tại

lf Chuyển con trở đến đầu trang sau

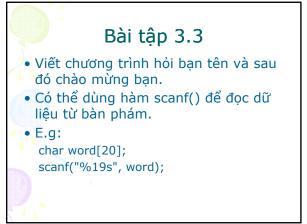
ln Chuyển con trở đến đầu dòng hiện tại.

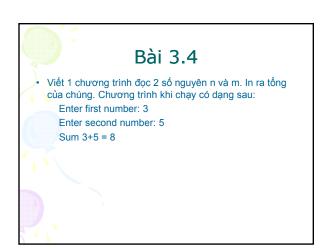
lt Chuyển con trở đến vị trị tab tiếp theo theo chiều ngạng

lv Chuyển con trở đến vị trị tab tiếp theo theo chiều dọc
```











#### Gợi ý

- Vì ngày trong tuần được xác định bởi 1 ký tự, bạn cần định nghĩa 1 tập ký tự, ví dụ 'm' cho Monday, 't' cho Tuesday, và 'h' for Thursday.
- Khi đọc 1 ký tự, dùng scanf(" %c",&day) để bỏ qua các dấu trống ở đầu.