

CONFIDENTIAL

## Week 3: Standard output introduction

For HEDSPI Project

## Vào/ra trong C

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

## Định dạng đầu ra với printf

- **printf**
  - Định dạng chính các đầu ra
    - Chỉ định sự chuyển đổi: cờ, độ rộng trường, độ chính xác, ...
  - Có thể làm tròn, đóng 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 đầu 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

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

## Ví dụ 1

```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf( "%d\n", 455 );
6     printf( "%i\n", 455 ); /*i same as d*/
7     printf( "%d\n", +455 );
8     printf( "%d\n", -455 );
9     printf( "%hd\n", 32000 );
10    printf( "%ld\n", 2000000000 );
11    printf( "%o\n", 455 );
12    printf( "%u\n", 455 );
13    printf( "%u\n", -455 );
14    printf( "%x\n", 455 );
15    printf( "%X\n", 455 );
16
17    return 0;
18 }
```

```
455
455
455
-455
32000
2000000000
707
455
65081
1c7
1C7
```

## Hiển thị số thực dấu phẩy động

- Dùng dấu chấm để phân cách (33.5)
- Biểu diễn dưới dạng mũ
  - $150.3 = 1.503 \times 10^2 = 1.503E+02$
- **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 dụ 2

```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf( "%e\n", 1234567.89 );
6     printf( "%e\n", +1234567.89 );
7     printf( "%e\n", -1234567.89 );
8     printf( "%E\n", 1234567.89 );
9     printf( "%f\n", 1234567.89 );
10    printf( "%g\n", 1234567.89 );
11    printf( "%G\n", 1234567.89 );
12
13    return 0;
14 }
```

```
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

- **%c**
  - In giá trị char
  - Không dùng để in ký tự đầu tiên của xâu
- **%s**
  - 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
- **Chú ý:**
  - Dùng dấu nhảy đơn cho ký tự ('\z')
  - Dùng dấu nhảy kép cho xâu ("z") (thực tế gồm 2 ký tự, 'z' và '\0')

## Ví dụ 3

```
1 #include <stdio.h>
2
3 int main()
4 {
5     char character = 'A';
6     char string[] = "This is a string";
7     const char *stringPtr = "This is also a string";
8
9     printf( "%c\n", character );
10    printf( "%s\n", "This is a string" );
11    printf( "%s\n", string );
12    printf( "%s\n", stringPtr );
13
14    return 0;
15 }
```

```
A
This is a string
This is a string
This is also a string
```

## Các dạng chuyển đổi khác

- **%p**
  - Hiện thị giá trị con trỏ (địa chỉ)
- **%n**
  - Lưu số ký tự đã được in ra bởi câu lệnh printf
  - Nhận tham số là con trỏ trỏ đến một số nguyên
  - Không hiển thị sử dụng %n
- **%%**
  - In dấu %

## Ví dụ 4

```
1#include <stdio.h>
2
3int main()
4{
5    int *ptr;
6    int x = 12345, y;
7
8    ptr = &x;
9    printf( "The value of ptr is %p\n", ptr );
10   printf( "The address of x is %p\n", &x );
11
12   printf( "Total characters printed on this line is: %n", &y );
13   printf( " %d\n", y );
14
15   y = printf( "This line has 28 characters\n" );
16   printf( " %d characters were printed\n", y );
17
18   printf( "Printing a % in a format control string\n" );
19
20   return 0;
21}
```

```
The value of ptr is 00639DF0
The address of x is 00639DF0
Total characters printed on this line is: 41
 28
This line has 28 characters
28 characters were printed
Printing a % in a format control string
```

## In với độ rộng trường và độ chính xác

- **Độ rộng trường**
  - Là kích thước của trường dữ liệu được in ra
- **Format**
  - Độ 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( "%.3f", 7, 2, 98.736 );`

## Ví dụ 5

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i = 873;
6     double f = 123.94536;
7     char s[] = "Happy Birthday";
8
9     printf( "Using precision for integers\n" );
10    printf( "\t%.4d\n\t%.9d\n", i, i );
11    printf( "Using precision for floating-point numbers\n" );
12    printf( "\t%.3f\n\t%.3e\n\t%.3g\n", f, f, f );
13    printf( "Using precision for strings\n" );
14    printf( "\t%.11s\n", s );
15
16    return 0;
17 }
```

```
Using precision for integers
0873
000000873

Using precision for floating-point numbers
123.945
1.239e+02
124

Using precision for strings
Happy Birth
```

## Flags trong kiểm soát dữ liệu in ra

- Flags
  - Đặt cờ ngay bên phải dấu %
  - Có thể kết hợp nhiều cờ

| Flag           | Description  |
|----------------|--|
| - (minus sign) | Left-justify the output within the specified field.  |
| + (plus sign)  | Display a plus sign preceding positive values and a minus sign preceding negative values.  |
| space          | Print a space before a positive value not printed with the + flag.   |
| #              | Prefix 0 to the output value when used with the octal conversion specifier o.  |
|                | Prefix 0x or 0X to the output value when used with the hexadecimal conversion specifiers x or X.   |
| .              | Force a decimal point for a floating-point number printed with e, E, f, g or G that does not contain a fractional part. (Normally the decimal point is only printed if a digit follows it.) For g and G specifiers, trailing zeros are not eliminated. |
| 0 (zero)       | Pad a field with leading zeros.  |

## Ví dụ 6

```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf( "%10s%10d%10c%10f\n", "hello", 7, 'a', 1.23 );
6     printf( "%-10s%-10d%-10c%-10f\n", "hello", 7, 'a', 1.23 );
7     return 0;
8 }
```

```
hello      7      a  1.230000
hello      7      a  1.230000
```

## Ví dụ 7

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int c = 1427;
6     double p = 1427.0;
7
8     printf( "%#o\n", c );
9     printf( "%#x\n", c );
10    printf( "%#X\n", c );
11    printf( "\n#g\n", p );
12    printf( "%#g\n", p );
13
14    return 0;
15 }
```

```
02623
0x593
0X593

1427
1427.00
```

## In các ký tự đặc biệt

| Cú pháp | Mô tả  |
|---------|--|
| '\'     | In dấu (')   |
| '\"'    | In dấu (")   |
| '\?'    | In dấu (?)   |
| '\\'    | In dấu (\)   |
| \a      | Tạo ra tiếng chuông hoặc thông báo dạng hình ảnh         |
| \b      | Chuyển con trỏ quay lại 1 vị trí ở dòng hiện tại         |
| \f      | Chuyển con trỏ đến đầu trang sau                         |
| \n      | Chuyển con trỏ đến đầu dòng sau.                         |
| \r      | Chuyển con trỏ đến đầu dòng hiện tại.                    |
| \t      | Chuyển con trỏ đến vị trí tab tiếp theo theo chiều ngang |
| \v      | Chuyển con trỏ đến vị trí tab tiếp theo theo chiều dọc   |

## Bài tập 3.1

- Viết chương trình hiển thị kích thước các kiểu dữ liệu cơ bản, ví dụ, int, long short, double, char...
- Có thể dùng hàm **sizeof** function để thực hiện việc này.
- Ví dụ: sizeof(int);

### Bài 3.2

- Viết chương trình sau. Dịch, kết nối và chạy nó.

```
#include<stdio.h>

void main ()
{
    int year;
    float height;

    year = 21;
    height = 1.77;

    printf("Ali is %d years old and %f meter height\n", year, height);
}
```

### Bài tập 3.3

- Viết chương trình hỏi bạn tên và sau đó chào mừng bạn.
- Có thể dùng hàm scanf() để đọc dữ liệu từ bàn phím.
- E.g:  
char word[20];  
scanf("%19s", word);

### Bài 3.4

- Viết 1 chương trình đọc 2 số nguyên n và m. In ra tổng của chúng. Chương trình khi chạy có dạng sau:  
Enter first number: 3  
Enter second number: 5  
Sum 3+5 = 8

### Bài 3.5

- Cửa hàng DVD của thư viện BK có 3 mức giá thuê:
- | Kiểu thuê | Giá/đĩa |
|-----------|---------|
| Qua đêm   | \$7.00  |
| 3 ngày    | \$5.00  |
| Tuần      | \$3.00  |
- Viết 1 chương trình nhập vào giá trị ngày trong tuần, số lượng đĩa khách hàng mượn qua đêm, 3 ngày, tuần. Chạy chương trình và in ra các giá trị để đảm bảo dữ liệu được đọc đúng.
- Sửa lại chương trình để tính tổng giá thuê DVDs

### 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.