




4. Vào ra dữ liệu (I/O)

🕒 Thời gian tạo	@August 27, 2023 5:33 PM
📄 Loại bài giảng	Lý thuyết

 Bài viết này sẽ sử dụng thuật ngữ I/O (Input/Output) thay cho "Vào ra dữ liệu"

1. I/O là gì?

- I/O hoặc Input/Output là sự giao tiếp giữa hệ thống xử lý thông tin (máy tính là ví dụ) với thế giới bên ngoài. Trong đó:
 - Đầu vào (input) là dữ liệu mà hệ thống nhận vào để xử lý.
 - Đầu ra (output) là dữ liệu thu được sau quá trình xử lý của hệ thống.

2. I/O của ngôn ngữ C

- Ngôn ngữ C sử dụng thư viện `<stdio.h>` (ở ngôn ngữ C++ có tên mới là `<cstdio>`)

- Trong thư viện này có 2 hàm (câu lệnh) chính để xử lý đầu vào/đầu ra, lần lượt là: `scanf` và `printf`.
- Hai hàm này sử dụng kỹ thuật "format" - định dạng dữ liệu, là việc quy các loại dữ liệu về một chuẩn nhất định.
- Ví dụ:

```
#include <stdio.h>

int main() {
    int age;

    printf("Enter your age: ");
    scanf("%d", &age);

    printf("Your age is %d.\n", age);

    return 0;
}
```

- Khi chạy chương trình:

```
Enter your age: 18
Your age is 18.
```

- Ở đoạn code trên, `%d` trong `scanf` và `printf` là ký hiệu đại diện cho dữ liệu kiểu `int`. Tùy vào các kiểu dữ liệu khác nhau thì các ký hiệu đại diện cũng sẽ khác nhau. Bởi lẽ mỗi một kiểu dữ liệu có khoảng giá trị khác nhau, cũng như cách biểu diễn khác nhau.
- Dưới đây là bảng tổng hợp các ký hiệu đại diện dùng trong `scanf` và `printf` của một số các kiểu dữ liệu khác nhau:

Kiểu dữ liệu (Data type)	Ký hiệu đại diện (Format specifier)
<code>char</code> , <code>signed char</code> , <code>unsigned char</code>	<code>%c</code>
<code>short</code> , <code>signed short</code>	<code>%hi</code> hoặc <code>%hd</code>
<code>unsigned short</code>	<code>%hu</code>
<code>int</code> , <code>signed int</code>	<code>%i</code> hoặc <code>%d</code>
<code>unsigned int</code>	<code>%u</code>
<code>long</code> , <code>signed long</code>	<code>%li</code> hoặc <code>%ld</code>
<code>unsigned long</code>	<code>%lu</code>
<code>long long</code> , <code>signed long long</code>	<code>%lli</code> hoặc <code>%lld</code>
<code>unsigned long long</code>	<code>%llu</code>
<code>float</code>	<code>%f</code>
<code>double</code>	<code>%lf</code>

- Tham khảo thêm: https://en.wikipedia.org/wiki/C_data_types
- `scanf` cho phép nhập nhiều biến cùng một lúc, bằng việc thêm các ký tự đại diện và các biến tương ứng, ví dụ:

```
scanf("%d%d%f", &a, &b, &c);
```

- Tương tự với hàm `printf`.

3. I/O của ngôn ngữ C++:

- Ngôn ngữ C++ cùng với sự xuất hiện của thư viện `<iostream>` là một bước cải tiến cho việc nhập xuất. Việc nhập xuất trở nên dễ dàng và đa dụng hơn rất nhiều.
- Thư viện này cung cấp 2 hàm (câu lệnh) `cin` và `cout`.
- Khi sử dụng 2 hàm này, chỉ cần chỉ ra những thứ cần được nhập vào/in ra, như đoạn kí tự, biến, ... mà không cần phải sử dụng ký hiệu đại diện.
- Ví dụ:

```
#include <iostream>

using namespace std;

int main() {
    int age;
    string name;

    cout << "Enter your name: ";
    cin >> name;

    cout << "Enter your age: ";
    cin >> age;

    cout << "Your name is " << name << " and you are " << age << " years old."

    return 0;
}
```

- Khi chạy chương trình:

```
Enter your name: Anh Dung
Enter your age: 18
Your name is Anh Dung and you are 18 years old.
```

- Với hàm `cin`, chúng ta sử dụng dấu `>>` để báo hiệu là cần nhập thêm biến gì. Ví dụ cho trường hợp nhập nhiều biến cùng lúc (các biến có thể thuộc các kiểu dữ liệu bất kỳ):

```
cin >> a >> b >> c >> d;
```

- Tương tự với hàm `cout` nhưng sử dụng dấu `<<`.
- So với 2 hàm `scanf` và `printf`, `cin` và `cout` dễ dùng hơn nhưng tốc độ chậm hơn.