

Các kiểu dữ liệu

- Các kiểu và định dạng chuyển đổi tương ứng
- Kiểu có dấu vs không dấu
- Kiểm tra kích thước kiểu
- Cách tính giá trị min max của 1 kiểu
- Hằng số
- Ví dụ minh họa



Các kiểu dữ liệu và định dạng chuyển đổi

Kiểu	Kích thước(byte)	Định dạng chuyển đổi
char	1	%c
short	>= 2	%hi
unsigned short	>= 2	%hu
int	>= 2	%d, %i
unsigned int	>= 2	%u
long	>= 4	%ld, %li
unsigned long	>= 4	%lu
long long	>= 8	%lld, %lli
unsigned long long	>= 8	%llu
float	4	%f, %g, %e, %a
double	8	%lf, %lg, %le, %la
long double	>= 8	%Lf, %Lg, %Le, %La
bool	1	Có thể dùng của kiểu số nguyên

Kiểu có dấu vs không dấu

- Kiểu có dấu(signed) có thể lưu trữ cả số âm và số dương
- Kiểu không dấu(unsigned) là kiểu chỉ lưu trữ được số ≥ 0
- Giá trị lớn nhất của kiểu không dấu gấp đôi kiểu có dấu



Kiểm tra kích thước của kiểu

- Kích thước của kiểu tùy thuộc hệ điều hành
- Sử dụng toán tử sizeof. Ví dụ:
- `sizeof(int)` có thể cho 2 hoặc 4



Cách tính giá trị min, max của 1 kiểu

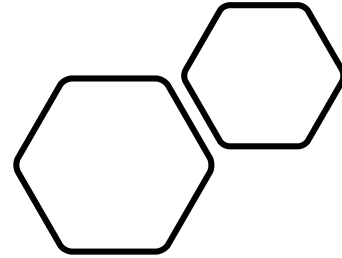
- Với kiểu có dấu:
 $\text{max} = 2^{(8 \cdot \text{size} - 1)} - 1;$
 $\text{min} = -(\text{max} + 1);$
- Với kiểu không dấu:
 $\text{min} = 0;$
 $\text{max} = 2^{(8 \cdot \text{size})} - 1;$
- Ví dụ, kiểu int 4 byte có giá trị max, min như sau:
 $\text{max} = 2^{(4 \cdot 8 - 1)} - 1 = 2147483647;$
 $\text{min} = -(\text{max} + 1) = -2147483648;$



Hằng số

- Là giá trị biết trước và không thay đổi
- Biến hằng số là biến mà giá trị của nó sau khi gán không thể thay đổi
- Tên hằng số viết hoa hoàn toàn
- Có 2 cách để tạo hằng số:
 - Dùng const: `const kiểu tên = giá_trị;`
 - Dùng #define: `#define tên giá_trị`





Tiếp theo

Các phép toán