



28TECH
Become A Better Developer



1. PROGRAM IN



KHUÔN MẪU CHUNG CỦA MỘT CHƯƠNG TRÌNH

INPUT



OUTPUT

CHƯƠNG TRÌNH C ĐẦU TIÊN

```
#include <stdio.h>

int main () {
    printf ("Hello 28tech!");
    return 0;
}
```

Khai báo thư viện cần thiết

Hàm main: là nơi bắt đầu thực thi của chương trình

Những phần chính của một chương trình C:

/01 Thư viện mà chương trình sử dụng. ✓

/02 Chương trình chính với mã nguồn.



NHỮNG CHÚ Ý KHI VIẾT CHƯƠNG TRÌNH C



Các câu lệnh **kết thúc bằng dấu ";"**



```
printf ("28tech")
```



```
printf ("28tech");
```



Luôn thực lễ các câu lệnh so với hàm main.



```
#include <stdio.h>
int main(){
printf("Hello 28tech");
return 0;
}
```



```
#include <stdio.h>
int main(){
    printf("Hello 28tech");
    return 0;
}
```



2. KIỂU DỮ LIỆU (DATA TYPE)



KIỂU DỮ LIỆU SỐ NGUYÊN

1 byte = 8 bit



Đối với số nguyên ta chia làm **số nguyên không dấu và có dấu**, từ số byte lưu trữ ta có thể suy ra **số bit** cần để biểu diễn số nguyên đó, quy tắc xác định giá trị của 1 số nguyên.

Giả sử số nguyên có K bit

- Số nguyên có dấu : -2^{K-1} tới $2^{K-1} - 1$
- Số nguyên không dấu : 0 tới $2^K - 1$



Kiểu dữ liệu số nguyên

Kiểu dữ liệu	Ý nghĩa	Số Byte	Giải giá trị có thể lưu
short	Số nguyên có dấu	2 byte	-32768 đến 32767
unsigned short	Số nguyên không dấu	2 byte	0 đến 65535
int	Số nguyên có dấu	4 byte	-2147483648 đến 2147483647
unsigned int	Số nguyên không dấu	4 byte	0 đến 4294967295
long long	Số nguyên	8 byte	-9223372036854775808 đến 9223372036854775807
unsigned long long	Số nguyên không dấu	8 byte	0 đến $2^{64} - 1$



KIỂU DỮ LIỆU SỐ THỰC

Kiểu dữ liệu	Ý nghĩa	Số Byte	Giải giá trị có thể lưu
float	Số thực độ chính xác đơn	4 byte	1.17549e-038 đến 3.40282e+038 Độ chính xác : 7 chữ số sau dấu phẩy
double	Số thực độ chính xác kép	8 byte	2.22507e-308 đến 1.79769e+308 Độ chính xác : 15 chữ số sau dấu phẩy

Kiểu dữ liệu ĐÚNG SAI

Kiểu dữ liệu	Ý nghĩa	Số Byte	Giải giá trị có thể lưu
int	Kiểu dữ liệu luận lý	4 byte	1 là đúng 0 là sai

KIỂU DỮ LIỆU KÝ TỰ

Kiểu dữ liệu	Ý nghĩa	Số Byte	Giải giá trị có thể lưu
char	Kiểu dữ liệu ký tự	1 byte	-128 tới 127

Tổng quan về kiểu dữ liệu:

/01 Kiểu số nguyên có **int** và **long long**.

>> Sử dụng long long cho các kết quả là số lớn

/02 Kiểu số thực có **float** và **double**.

>> Sử dụng double vì có độ chính xác cao

/03 Kiểu đúng sai dùng kiểu **int**

/04 Kiểu ký tự có **char**.



ĐẶC TẢ CÁC KIỂU DỮ LIỆU CHÍNH



Trong ngôn ngữ lập trình C, bạn cần nắm được đặc tả của các kiểu dữ liệu. Đặc tả đặc biệt quan trọng vì nó giúp các bạn có thể nhập giá trị cho biến từ bàn phím và in ra giá trị của biến lên màn hình.

Kiểu dữ liệu	Đặc tả tương ứng
int	%d
long long	%lld
float	%f
double	%lf
char	%c





3. BIẾN (VARIABLE)



BIẾN (VARIABLE)



Biến được sử dụng để **lưu các giá trị** trong quá trình tính toán của chương trình. Tùy theo kiểu dữ liệu của biến, một ô trong bộ nhớ sẽ được cấp phát để lưu trữ giá trị của biến này.

SYNTAX

```
Data_Type Variable_Name;
```

Ví dụ:

```
int x; long long b;  
char ki_tu; int check;  
double dienTich;
```



BIẾN (VARIABLE)

Quy tắc đặt tên biến:

Ví dụ cách đặt sai:

01	Không đặt tên biến bắt đầu bằng chữ số	1dientich, 2chuvi, 222bankinh, 9a,...
02	Tên biến không được chứa dấu cách và các kí tự đặc biệt	ban kinh, dien#tich, chu@vi,...
03	Tên biến không được trùng với tên từ khóa trong C	int, for, while, ...
04	Tên biến trong C là phân biệt hoa thường	banKinh và BanKinh là 2 biến khác nhau
05	Không được đặt 2 biến có cùng tên trong cùng một phạm vi	int a; float a;



4. CHÚ THÍCH TRONG C

//

/* */



CHÚ THÍCH (COMMENT)



Chú thích là **một giải pháp bổ sung thông tin** vào code của bạn, nhằm làm rõ nội dung, giải thích câu lệnh, mục đích của code,...



Giúp người đọc code có thể **dễ nắm bắt** nội dung code và thuận lợi trong việc bảo trì code.



Các chú thích sẽ **không được coi là câu lệnh** và sẽ được loại bỏ khi chương trình thực thi.



CHÚ THÍCH (COMMENT)

Để chú thích trên một dòng, ta dùng //

```
#include <stdio.h>
```

```
//Đây là chú thích
```

```
//Chú thích giúp code rõ ràng hơn
```

```
int main(){
```

```
    //Câu lệnh này in ra dòng chữ hello world !
```

```
    printf("hello world !");
```

```
    return 0;
```

```
}
```

CHÚ THÍCH (COMMENT)

Để chú thích trên nhiều dòng, ta dùng `/* ... */`

```
#include <stdio.h>
```

```
int main(){  
    /* Câu lệnh này in ra  
    dòng chữ hello world !  
    */  
    printf("hello world !");  
    return 0;  
}
```



MỘT SỐ VÍ DỤ

Khai báo biến

In giá trị của biến ra màn hình

Nhập giá trị cho biến từ bàn phím

MỘT SỐ VÍ DỤ

Khai báo biến

```
#include <stdio.h>

int main(){
    int a;    // biến a có kiểu int
    int x, y, z; // khai báo 3 biến cùng kiểu int
    long long dien_tich; // biến dien_tich có kiểu ll
    int check; // biến check kiểu đúng sai
    char kitu; // biến kitu kiểu char
}
```

- **Chú ý:** Khi chỉ **khai báo biến** mà **không khởi tạo giá trị** cho nó thì giá trị của nó là những giá trị không xác định, thường gọi là **giá trị rác**.



MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

Khi muốn in giá trị của biến số ra màn hình, ta **kết hợp câu lệnh printf và đặc tả của biến**. Bạn cần **khai báo thư viện stdio.h** để sử dụng lệnh printf này. Bạn cũng có thể thêm các câu dẫn cho hàm printf để chương trình rành mạch hơn.

```
printf ([Đặc tả], Tên_biến);
```

Phần đặc tả được đưa vào trong nháy kép, các bạn có thể dùng hàm printf để in ra 1 hoặc nhiều giá trị của nhiều biến cùng 1 lúc.



MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

```
#include <stdio.h>
```

```
int main(){  
    int a = 100; // Biến a sẽ không còn giá trị  
                //rác nữa, mà có giá trị là 100  
    printf("Gia tri cua a la : %d\n", a);  
    int x = 1, y = 2, z = 3;  
    printf("%d %d %d\n", x, y, z);  
    return 0;  
}
```

Output:

```
Gia tri cua a la : 100  
1 2 3
```


MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

```
#include <stdio.h>
//Hãy chú ý sử dụng đúng đặc tả
int main(){
    long long tech28 = 1019129395468;
    printf("%lld\n", tech28);
    long long a = 64812156456948946;
    printf("%d\n", a); // ?????????
    return 0;
}
```

Output:

```
1019129395468
849600741
```



MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

```
#include <stdio.h>
int main(){
    char kitu = 'A';
    printf("%c\n", kitu);
    char kt1 = 't', kt2 = 'e', kt3 = 'c', kt4 = 'h';
    printf("%c%c%c%c\n", kt1, kt2, kt3, kt4);
    printf("%c %c %c %c\n", kt1, kt2, kt3, kt4);
    return 0;
}
```

Output : A

tech

t e c h

MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

Chú ý khi in số kiểu float và double, bạn cần chỉ định số lượng số thập phân đằng sau dấu phẩy

float

```
printf ("%xf, Tên_biến);
```

double

```
printf ("%xlf, Tên_biến);
```

Với x là số lượng số thập phân sau dấu phẩy muốn in ra



MỘT SỐ VÍ DỤ

Hiển thị giá trị của biến ra màn hình

```
#include <stdio.h>
```

```
int main(){  
    float a = 3.91823123;  
    printf("%.2f\n", a);  
    double b = 3.192391239193;  
    printf("%.12lf\n", b);  
    return 0;  
}
```

Output :

3.92

3.192391239193



MỘT SỐ VÍ DỤ

Nhập giá trị cho biến từ bàn phím

Để nhập giá trị cho biến từ bàn phím thay vì khởi tạo thủ công như các ví dụ trên, ta **sử dụng hàm scanf** kèm theo **đặc tả của biến** mà bạn muốn nhập. Bạn cần **thêm thư viện stdio.h** vào chương trình để sử dụng hàm scanf.

```
scanf ([Đặc tả], &Tên_biến);
```

Phần đặc tả được đưa vào trong nháy kép, bạn có thể nhập giá trị cho 1 biến hoặc nhiều biến với chỉ một lần dùng hàm scanf.



MỘT SỐ VÍ DỤ

Nhập giá trị cho biến từ bàn phím

```
scanf ([Đặc tả], &Tên_biến);
```

Dấu & trước tên biến **thể hiện toán tử địa chỉ**, có thể hiểu đơn giản, khi bạn muốn dùng hàm scanf để gán giá trị cho biến từ bàn phím **hàm scanf cần phải biết địa chỉ của biến** này trong bộ nhớ và **tìm đến ô nhớ này để gán giá trị** cho nó.

Chú ý : Khi nhập xong giá trị các bạn ấn enter để cin bắt đầu nhập giá trị cho biến, trong trường hợp bạn nhập giá trị cho nhiều số, có thể nhập từng số rồi enter hoặc nhập 1 lúc nhiều số rồi ấn enter 1 lần.

MỘT SỐ VÍ DỤ

Nhập giá trị cho biến từ bàn phím

```
#include <stdio.h>
```

```
int main(){  
    int a;  
    printf("Nhap gia tri cho bien a :"); // Câu dẫn  
    scanf("%d", &a);  
    printf("Gia tri cua a vua nhap : %d\n", a);  
    int x, y, z;  
    // Dùng 1 hàm scanf để nhập giá trị cho 1 lúc 3 biến  
    scanf("%d %d %d", &x, &y, &z);  
    printf("%d %d %d", x, y, z);  
    return 0;  
}
```



MỘT SỐ VÍ DỤ

Nhập giá trị cho biến từ bàn phím

- **Chú ý:** Khi nhập giá trị cho nhiều biến cùng 1 lúc, giữa các đặc tả có thể có dấu cách nhưng sau đặc tả cuối cùng không được có dấu cách.

- **Ví dụ sai:**

```
#include <stdio.h>
```

```
int main(){  
    int x, y, z;  
    // Dùng 1 hàm scanf để nhập giá trị cho 1 lúc 3 biến  
    scanf("%d %d %d ", &x, &y, &z);  
    printf("%d %d %d", x, y, z);  
    return 0;  
}
```



MỘT SỐ VÍ DỤ

Nhập giá trị cho biến từ bàn phím

```
#include <stdio.h>
```

```
int main(){  
    float a;  
    double b;  
    char c;  
    scanf("%f %lf %c", &a, &b, &c);  
    printf("%.2f %.5lf %c\n", a, b, c);  
    return 0;  
}
```

Input : 2.123 5.234 @

Output : 2.12 5.23400 @