

Kiểu Cấu Trúc (STRUCT)



Giới thiệu về struct:



Cấu trúc hay struct trong ngôn ngữ lập trình C cho phép bạn tự định nghĩa kiểu dữ liệu mới cho mình, nhằm giải quyết các bài toán khi mà thông tin đối tượng bạn cần lưu trữ không thể chỉ sử dụng kiểu dữ liệu cơ bản.



Để lưu thông tin của một sinh viên với tên, lớp, điểm gpa... thì việc xây dựng một kiểu dữ liệu có thể lưu toàn bộ thông tin của sinh viên này là cần thiết.



1. Khai báo struct trong C:

CÚ PHÁP:

```
struct struct_name{  
    //Danh sách các thuộc tính  
};
```

- Danh sách các thuộc tính ở đây chính là các trường dữ liệu để lưu thông tin cho struct. Bạn phải chỉ ra kiểu dữ liệu của các trường dữ liệu này.

VD: Cấu trúc sinh viên có tên, lớp, điểm gpa

```
struct SinhVien{  
    char ten[100];  
    char lop[100];  
    double gpa;  
};
```

1. Khai báo struct trong C:



Trong ngôn ngữ lập trình C thì struct chưa được sử dụng để khai báo như một kiểu dữ liệu thông thường mà phải có thêm từ khóa struct. Để có thể sử dụng struct như một kiểu dữ liệu, ta nên typedef cho struct trước.

Ví dụ:

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

struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};

int main(){
    struct SinhVien v;
}
```

Sử dụng typedef:

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

struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};

typedef struct SinhVien SinhVien;

int main(){
    SinhVien v;
}
```



2. Truy cập vào các thành phần của struct:

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

```
struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};
```

```
typedef struct SinhVien SinhVien;
```

```
int main(){
    SinhVien v;
    gets(v.ten);
    gets(v.lop);
    scanf("%lf", &v.gpa);
    printf("Thong tin SV : %s %s %.2lf\n", v.ten, v.lop, v.gpa);
}
```



Để truy cập vào các trường dữ liệu của struct ta dùng **toán tử '.'**



3. Gán 2 biến struct:

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

```
struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};
```

```
typedef struct SinhVien SinhVien;
```

```
int main(){
    SinhVien v;
    gets(v.ten); gets(v.lop);
    scanf("%lf", &v.gpa);
    printf("Thong tin SV : %s %s %.2lf\n", v.ten, v.lop, v.gpa);
    SinhVien t = v;
    printf("Thong tin SV : %s %s %.2lf\n", t.ten, t.lop, t.gpa);
}
```



Khi bạn gán 2 biến struct cho nhau, mọi trường dữ liệu của 2 biến này có nội dung giống nhau.



4. Mảng struct:



Sau khi xây dựng struct và typedef, bạn có thể sử dụng kiểu struct như một kiểu dữ liệu nguyên thủy.



Ví dụ để quản lý 1 danh sách sinh viên có không quá 1000 người bạn có thể khai báo 1 mảng SinhVien có tối đa 1000 phần tử.



4. Mảng struct:

Mã nguồn nhập và in thông tin N sinh viên từ bàn phím:

Hàm nhập thông tin

```
SinhVien nhap(){  
    SinhVien x;  
    printf("Nhap ten :");  
    gets(x.ten);  
    printf("Nhap lop :");  
    gets(x.lop);  
    printf("Nhap gpa :");  
    scanf("%lf", &x.gpa);  
    return x;  
}
```

Hàm xuất thông tin

```
void xuat(SinhVien x){  
    printf("Ho ten : %s\n", x.ten);  
    printf("Lop : %s\n", x.lop);  
    printf("GPA : %.2lf\n", x.gpa);  
}
```


4. Mảng struct:

Mã nguồn nhập và in thông tin N sinh viên từ bàn phím:

Hàm main:

```
int main(){
    SinhVien a[1000]; int N;
    printf("Nhap so luong sinh vien :");
    scanf("%d", &N);
    for(int i = 0; i < N; i++){
        getchar(); // ???
        a[i] = nhap();
    }
    for(int i = 0; i < N; i++){
        xuat(a[i]);
    }
}
```

Input / Output

```
Nhap so luong sinh vien :3
Nhap ten :Tran Xuan Loc
Nhap lop :CNTT1
Nhap gpa :3.1
Nhap ten :Hoang Dinh Nam
Nhap lop :CNTT2
Nhap gpa :2.8
Nhap ten :Hoang Nhat Long
Nhap lop :DTVT1
Nhap gpa :3.6
Ho ten : Tran Xuan Loc
Lop : CNTT1
GPA : 3.10
Ho ten : Hoang Dinh Nam
Lop : CNTT2
GPA : 2.80
Ho ten : Hoang Nhat Long
Lop : DTVT1
GPA : 3.60
```

5. Con trỏ kiểu struct:

Tương tự như kiểu dữ liệu thông thường, bạn có thể **khai báo con trỏ struct**. Nhưng **khi truy cập vào các trường dữ liệu của biến con trỏ struct ta dùng -> thay vì dùng toán tử '.'**

Ví dụ:

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

struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};

typedef struct SinhVien SinhVien;

int main(){
    SinhVien x;
    strcpy(x.ten, "28tech");
    strcpy(x.lop, "CNTT1");
    x.gpa = 2.5;
    SinhVien *ptr = &x;
    printf("%s %s %.2lf\n", ptr->ten, ptr->lop, ptr->gpa);
}
```

OUTPUT

28tech CNTT1 2.50



5. Con trỏ kiểu struct:

Nhập thông tin cho 1 biến struct bằng hàm truyền tham chiếu

Ví dụ:

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

struct SinhVien{
    char ten[100];
    char lop[100];
    double gpa;
};

typedef struct SinhVien SinhVien;

void nhap(SinhVien *x){
    gets(x->ten);
    gets(x->lop);
    scanf("%lf", &x->gpa);
}

int main(){
    SinhVien x;
    nhap(&x);
    printf("%s %s %.2lf\n", x.ten, x.lop, x.gpa);
}
```

5. Con trỏ kiểu struct:

Quiz: Hãy chỉ ra lỗi của chương trình

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

struct SinhVien{
    char *ten;
    char *lop;
    double gpa;
};

typedef struct SinhVien SinhVien;

int main(){
    SinhVien x;
    gets(x.ten);
    gets(x.lop);
    scanf("%lf", x.gpa);
}
```