Chào mừng các bạn trở lại với Series Lập Trình Căn Bản C. Hôm nay chúng ta sẽ học về Struct trong ngôn ngữ C.

Struct hay có tên gọi khác là Kiểu Cấu Trúc.

Lại một lần nữa chúng ta nói về biến.

Chúng ta đã biết rằng có rất nhiều loại biến, và nhiều kiểu dữ liệu: int, float, double, nguyên, thực,..

Và khi học đến struct, chúng ta sẽ biết nó với tên gọi Kiểu Dữ Liệu Người Dùng.

Nó là một kiểu dữ liệu do chúng ta tạo ra và phục vụ cho riêng chúng ta.

**STRUCT**

1. Cú pháp:

Typedef struct{

Biến\_thành\_viên

}Tên\_struct;

Ví dụ 1:

Typedef struct{

Int tu, mau;

} PhanSo;

Struct này là struct phân số có 2 biến thành viên là tu kiểu int và mau kiểu int.

Ví dụ 2:

Typedef struct{

Char hoten[50];

Int tuoi;

}HocSinh;

Struct có tên là HocSinh, có 2 biến thành viên là chuỗi họ tên và tuoi kiểu int.

Tương tự như vậy, các bạn có thể tạo ra nhiều struct của riêng mình.

1. Cách sử dụng

Khi đã có một biến riêng mình rồi, việc tiếp theo bạn sẽ quan tâm là **làm sao để sử dụng nó**?

Đầu tiên, để khai báo nó, chúng ta sẽ khai báo như sau:

**Ten\_struct ten\_bien;**

Ví dụ:

PhanSo a; // biến a kiểu PhanSo

PhanSo a[10]; // mảng 10 PhanSo

HocSinh a; // biến a kiểu HocSinh

HocSinh a[10]; // mảng 10 học sinh.

Sau khi biết cách khai báo rồi thì chúng ta sẽ quan tâm tới việc sử dụng nó.

Sử dụng struct và các thành viên struct theo cú pháp sau đây:

Tên\_biến\_struct dấu\_chấm tên\_biến\_thành\_viên;

Ví dụ:

#include "stdio.h"

typedef struct{

int tu, mau;

}PhanSo;

int main(){

PhanSo a;

a.tu = 5;

a.mau = 2;

printf("%d / %d", a.tu, a.mau);

return 0;

}

Ví dụ 3: Tạo struct ngày tháng năm

#include "stdio.h"

typedef struct{

int ngay, thang, nam;

}NgayThang;

int main(){

NgayThang a;

printf("Nhap ngay - thang - nam: ");

scanf("%d %d %d", &a.ngay, &a.thang, &a.nam);

printf("%d - %d - %d", a.ngay, a.thang, a.nam);

return 0;

}

Vậy là ta có thể truy xuất và sử dụng các biến thành viên được rồi. Giờ ta sẽ đến với Mảng Struct.

**Mảng struct**

Tương tự như việc cài đặt một struct bình thường, ta sẽ thử với một mảng struct.

#include "stdio.h"

#include "stdlib.h"

typedef struct{

char hoTen[50];

int tuoi;

}DanhSach;

int main(){

DanhSach a[10];

int i;

printf("Nhap danh sach\n");

printf("----------------\n");

for(i = 0 ; i < 10; i++){

printf("Thanh vien thu %d\n", i+1);

printf("Ten: ");

fflush(stdin);

gets(a[i].hoTen);

printf("Tuoi: ");

scanf("%d", &a[i].tuoi);

}

printf("----------------\n");

printf("%-30s %5s\n", "Ho ten", "Tuoi");

for(i = 0 ; i < 10 ;i++){

printf("%-30s %5d\n", a[i].hoTen, a[i].tuoi);

}

return 0;

}

Cách truy xuất của nó hoàn toàn tương tự, để truy xuất phần tử tại vị trí nào, ta thêm chỉ số của nó vào giống như mảng.

**Struct lồng nhau**

Struct có thể lồng nhau được, các bạn theo dõi code sau:

#include "stdio.h"

#include "stdlib.h"

typedef struct{

int ngay,thang,nam;

}NgayThang;

typedef struct{

char hoTen[50];

int tuoi;

NgayThang date;

}DanhSach;

int main(){

DanhSach a[10];

int i;

printf("Nhap danh sach\n");

printf("----------------\n");

for(i = 0 ; i < 2; i++){

printf("Thanh vien thu %d\n", i+1);

printf("Ten: ");

fflush(stdin);

gets(a[i].hoTen);

printf("Tuoi: ");

scanf("%d", &a[i].tuoi);

printf("Nhap ngay - thang - nam: ");

scanf("%d %d %d", &a[i].date.ngay, &a[i].date.thang, &a[i].date.nam);

}

printf("----------------\n");

printf("%-30s %-5s %5s\n", "Ho ten", "Tuoi", "Sinh Nhat");

for(i = 0 ; i < 2 ;i++){

printf("%-30s %-5d %d-%d-%d\n", a[i].hoTen, a[i].tuoi, a[i].date.ngay, a[i].date.thang, a[i].date.nam);

}

return 0;

}

**Con trỏ Struct**

Đối với con trỏ struct, khi truy xuất nó chúng ta không sử dụng dấu chấm (.) mà sử dụng dấu truy xuất (->)

Ví dụ:

#include "stdio.h"

typedef struct{

int a, b;

}test;

int main(){

test \*t = (test \*) malloc (1 \* sizeof(test \*));

printf("Nhap a: ");

scanf("%d", &t->a);

printf("Nhap b: ");

scanf("%d", &t->b);

printf("a = %d, b = %d", t->a, t->b);

return 0;

}

Bài giảng struct của chúng ta đến đây là kết thúc, và cũng đã kết thúc Series Lập Trình C Căn Bản.

Phần này cũng sẽ có những bài tập, nhưng mình sẽ chuyển nó sang ở mục Giải Bài Tập Series C.

Hi vọng những bài học của mình có thể đem đến cho các bạn những kiến thức mới, bổ ích. Xin chân thành cảm ơn các bạn.