

CHUYÊN ĐỀ 5: CÁC KIỂU DỮ LIỆU NÂNG CAO – SẮP XẾP

Mục tiêu

- ✓ Tìm hiểu kiểu dữ liệu cấu trúc và công dụng
- ✓ Định nghĩa cấu trúc
- ✓ Khai báo các biến kiểu cấu trúc
- ✓ Cách truy cập vào các phần tử của cấu trúc
- ✓ Khởi tạo biến cấu trúc
- ✓ Sử dụng biến cấu trúc trong câu lệnh gán
- ✓ Cách truyền tham số cấu trúc
- ✓ Sử dụng mảng các cấu trúc
- ✓ Tìm hiểu cách khởi tạo mảng các cấu trúc
- ✓ Con trỏ cấu trúc
- ✓ Cách truyền tham số kiểu con trỏ cấu trúc
- ✓ Tìm hiểu từ khóa typedef
- ✓ Sắp xếp mảng bằng phương pháp Bubble sort và Insertion sort.

Kiến thức cần nắm vững

- ✓ Một cấu trúc là tập các biến có thể có kiểu dữ liệu khác nhau được nhóm lại với nhau dưới cùng một tên
- ✓ Định nghĩa cấu trúc sẽ tạo ra kiểu dữ liệu mới cho phép người dùng sử dụng chúng để khai báo các biến kiểu cấu trúc
- ✓ Toán tử chấm (.), hay còn được gọi là toán tử thành viên
- ✓ Có thể có một cấu trúc nằm trong một cấu trúc khác
- ✓ Một biến cấu trúc có thể được truyền vào một hàm như là một tham số
- ✓ Cách sử dụng thông dụng nhất của cấu trúc là dưới hình thức mảng cấu trúc
- ✓ Toán tử -> được sử dụng để truy cập vào các phần tử của một cấu trúc thông qua một con trỏ trỏ đến cấu trúc đó
- ✓ Một kiểu dữ liệu mới có thể được định nghĩa bằng từ khóa typedef
- ✓ Hai phương pháp dùng để sắp xếp một mảng là bubble sort và insertion sort
- ✓ Trong bubble sort, giá trị của các phần tử được so sánh với giá trị của phần tử kế tiếp. Trong phương pháp này, các phần tử nhỏ hơn nổi lên dần, và cuối cùng mảng sẽ được sắp xếp
- ✓ Trong insertion sort, ta xét mỗi phần tử trong mảng và chèn vào vị trí đúng của nó giữa các phần tử đã được sắp xếp

- ✓ Việc định nghĩa cấu trúc sẽ tạo ra kiểu dữ liệu mới cho phép người dùng sử dụng chúng để khai báo các biến kiểu cấu trúc.
- ✓ Các biến trong cấu trúc được gọi là các phần tử của cấu trúc hay thành phần của cấu trúc.

Khai báo một biến cấu trúc : **struct cat books1;**

Ví dụ:

```
struct book{  
    char bk_name[25];  
    char author[20];  
    int edn;  
    float price;  
};
```



Bài tập hướng dẫn

Bài 1: Viết chương trình cho phép người dùng nhập vào thông tin của 2 đến 5 Sinh viên bao gồm Mã Sinh viên, Tên Sinh viên, Điểm trung bình, Ngày, tháng, năm Sinh sau đó hiển thị ra màn hình (Sử dụng mảng cấu trúc).

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

/*
 *
 */

struct students {
    char Code[20]; // ma sinh vien
    char fullName[30]; // ho ten
    double markAvg; // diem trung

    struct ngaysinh {
        int day, month, years;
    } birthday;
};

int main(int argc, char** argv) {
    int n = 2, i;
    struct students infoStudent[n];
    for (i = 0; i < n; i++) {
#define st infoStudent[i]
        printf("Nhap du lieu cho sinh thu %d: \n", i + 1);
        printf("Code: ");
        gets(st.Code);
        printf("Ho ten: ");
        gets(st.fullName);
        printf("DiemTB: ");
        scanf("%lf", &st.markAvg);
        printf("ngay sinh: ");
        scanf("%d/%d/%d", &st.birthday.day, &st.birthday.month, &st.birthday.years);
        fflush(stdin);
    }
    printf("\n-----Thong tin sinh vien-----\n");
    printf("%-20s%-30s%-7s%-10s\n", "Ma", "Ho ten", "Diem tb", "Ngay sinh");
    for (i = 0; i < n; i++) {
#define st infoStudent[i]
        printf("%-20s%-30s%-7.2lf%-2d/%2d/%4d\n", st.Code, st.fullName, st.markAvg,
            st.birthday.day, st.birthday.month, st.birthday.years);
    }

    return (EXIT_SUCCESS);
}
```

```
Nhap du lieu cho sinh thu 1:
Code: s1
Ho ten: Nguyen Van A
DiemTB: 9.5
ngay sinh: 4/5/1998
Nhap du lieu cho sinh thu 2:
Code: s2
Ho ten: Nguyen Van B
DiemTB: 8.7
ngay sinh: 6/5/1998

-----Thong tin sinh vien-----
Ma          Ho ten          Diem tbNgay sinh
s1          Nguyen Van A    9.50    4/ 5/1998
s2          Nguyen Van B    8.70    6/ 5/1998
Press [Enter] to close the terminal ...
```

Bài 2: Sử dụng hàm viết chương trình chương trình nhập xuất Sinh viên bao gồm mã Sinh Viên, Tên Sinh Viên, điểm Toán, Lý, Hóa, Anh, Văn, DTB. Sắp xếp Sinh viên có điểm trung bình thấp đến cao. Sinh viên có điểm trung bình từ 7-9. Tìm kiếm dựa vào tên Sinh viên. In ra màn hình các kết quả.

```
1  #include <stdio.h>
2  #include <conio.h>
3  #include <string.h>
4  #define MAX 50
5
6  typedef struct sinhvien
7  {
8      int masv;
9      char ten[30];
10     float toan, ly, hoa, van, anh, DTB;
11 }Kieuhinhvien;
12
13 void nhap(Kieuhinhvien a[MAX], int n)
14 {
15     int i;
16     for(i = 0; i < n; i++)
17     {
18         printf("\nNhap sinh vien thu %d: ", i+1);
19         fflush(stdin);
20         printf("\nNhap ten sinh vien: ");
21         gets(a[i].ten);
22         fflush(stdin);
23         printf("\nNhap ma sinh vien: ");
24         scanf("%d", &a[i].masv);
25         fflush(stdin);
26         printf("\nNhap diem Toan, Ly, Hoa, Van, Anh, voi dk 0-10: ");
27         scanf("%f %f %f %f %f", &a[i].toan, &a[i].ly, &a[i].hoa, &a[i].van, &a[i].anh);
28         fflush(stdin);
29         a[i].DTB = (a[i].toan + a[i].ly + a[i].hoa + a[i].van + a[i].anh)/5;
30         fflush(stdin);
31     }
32 }
```

```

34 void xuat(Kieuhinhvien a[MAX], int n)
35 {
36     int i;
37     printf("%-20s %-30s %-7s %-7s %-7s %-7s %-7s %-7s", "Hoten", "MaSV", "Toan", "Ly", "Hoa", "Anh", "Van", "DTB");
38     for(i = 0; i < n; i++)
39     {
40         printf("\n%-20s %-30d %-7.1f %-7.1f %-7.1f %-7.1f %-7.1f %-7.1f",
41             a[i].ten, a[i].masv, a[i].toan, a[i].ly, a[i].hoa, a[i].van, a[i].anh, a[i].DTB);
42     }
43 }

```

```

44 void hoanvi(float x, float y)
45 {
46     float temp = x;
47     x = y;
48     y = temp;
49 }
50 void sapxep(Kieuhinhvien a[MAX], int n)
51 {
52     int i, j;
53     for(i = 0; i < n-1; i++)
54     {
55         for(j = i+1; j < n; j++)
56         {
57             if(a[i].DTB > a[j].DTB)
58             {
59                 hoanvi(a[i].DTB, a[j].DTB);
60             }
61         }
62     }
63 }
64
65 void thongke(Kieuhinhvien a[MAX], int n)
66 {
67     int i;
68     for(i = 0; i < n; i++)
69     {
70         if(a[i].DTB >= 7 && a[i].DTB <= 9)
71         {
72             printf("\n%-2s %5d %8.1f %8.1f %8.1f %8.1f %8.1f %8.1f",
73                 a[i].ten, a[i].masv, a[i].toan, a[i].ly, a[i].hoa, a[i].van, a[i].anh, a[i].DTB);
74         }
75     }
76 }
77

```



```

77 void timkiem(Kieuhvien a[MAX], int n, char tim[MAX])
78 {
79     int i;
80     printf("\nNhap vao ten sinh vien: "); gets(tim);
81     for(i = 0; i < n; i++)
82     {
83         if(strcmp(tim,a[i].ten)==0)
84         {
85             printf("\n%2s %5d %8.1f %8.1f %8.1f %8.1f %8.1f %8.1f\n",
86                 a[i].ten, a[i].masv, a[i].toan, a[i].ly, a[i].hoa, a[i].van, a[i].anh, a[i].DTB);
87         }
88     }
89 }
90
91 }
92 int main()
93 {
94     struct sinhvien a[MAX];
95     int n;
96     char tim[MAX];
97     printf("\nNhap vao so sinh vien n = ");
98     scanf("%d", &n);
99     nhap(a, n);
100    xuat(a,n);
101    printf("\nSap xep sinh vien tang dan: \n\n");
102    sapxep(a, n);
103    xuat(a,n);
104    printf("\nSinh vien co dien TB tu 7 den 9: \n\n");
105    thongke(a, n);
106    printf("\ntim kiem sinh vien \n\n");
107    timkiem(a, n, tim);
108
109    getch();
110 }

```

```

Nhap sinh vien thu 2:
Nhap ten sinh vien: Nguyen Van B

Nhap ma sinh vien: 2

Nhap diem Toan, Ly, Hoa, Van, Anh, voi dk 0-10: 6
7
8
9
4

```

Hoten	MaSV	Toan	Ly	Hoa	Anh	Van	DTB
Nguyen Van A	1	5.0	4.0	2.0	6.0	7.0	4.8
Nguyen Van B	2	6.0	7.0	8.0	9.0	4.0	6.8

```

Sap xep sinh vien tang dan:

Hoten      MaSV      Toan  Ly   Hoa  Anh  Van  DTB
Nguyen Van A  1      5.0   4.0   2.0   6.0   7.0   4.8

Nguyen Van B  2      6.0   7.0   8.0   9.0   4.0   6.8

Sinh vien co dien TB tu 7 den 9:

tim kiem sinh vien

Nhap vao ten sinh vien: Nguyen Van A

Nguyen Van A  1      5.0   4.0   2.0   6.0   7.0   4.8

```

Bài tập tự làm

Bài 1: Viết chương trình cho phép người dùng nhập vào Mã Sinh viên, Họ tên Sinh viên, điểm các môn excel, word, powpoint. Khi nào người dùng nhập vào mã Sinh viên là “END” kết thúc nhập thông tin Sinh viên và hiển thị thông tin vừa nhập ra màn hình.(lưu ý sử dụng dùng hàm).

Bài 2: Cho một danh sách sinh viên được mô tả chi tiết như sau:

- Danh sách sinh viên gồm n sinh viên. với n được nhập bất kỳ.
- Thông tin một sinh viên bao gồm: Số thứ tự, Mã số sinh viên, Họ tên sinh viên, Điểm Nhập môn lập trình, Môn toán A1, Môn toán A2, Môn Vật lý kỹ thuật, Môn anh văn, Điểm trung bình tích lũy, Xếp loại (xuất sắc, giỏi, khá, trung bình, yếu).
- Thông tin môn học bao gồm: Mã môn, Tên môn, Số tín chỉ, Điểm.

Thực hiện các yêu cầu sau:

- Hãy thiết kế và xây dựng các cấu trúc (struct) phù hợp cho bài toán
- Xây dựng một chương trình hoàn chỉnh cho một lớp học đảm bảo các thao tác sau:

- Hàm nhập xuất danh sách sinh viên
- Tìm một sinh viên thông qua mã số sinh viên hoặc qua tên sinh viên
- Tìm sinh viên có điểm trung bình cao nhất
- Sắp xếp danh sách sinh viên tăng dần .
- + Theo điểm trung bình tích lũy
- + Hoặc Theo tên theo thứ tự ABC ...

Lưu ý:

- ĐTBTC là trung bình cộng của tất cả các môn dựa trên điểm số và số tín chỉ.
- Lưu ý: Xếp loại được tính như sau:
 - + Xuất sắc: $\text{ĐTBTL} \geq 9$
 - + Giỏi với: $8.0 \leq \text{ĐTBTL} < 9$
 - + Khá với: $6.5 \leq \text{ĐTBTL} < 8.0$
 - + Trung bình: $5.0 \leq \text{ĐTBTL} < 6.5$
 - + Yếu: $\text{ĐTBTL} < 5$