


<b>Trường ĐH CNTP TP.HCM</b> <b>Khoa: CNTT</b> <b>Bộ môn: CNPM</b> <b>Môn: TH NMLT</b>	<b>BÀI 8. FILE DỮ LIỆU &amp; CẤU TRÚC (STRUCT)</b>	
---	--	---

### A. MỤC TIÊU:

- Khai báo được biến kiểu mở file, đóng file.
- Trình bày được cách đọc một số nguyên, cách đọc một dòng dữ liệu lưu trong file.
- Phân tích được tình huống sử dụng file để đọc, lưu dữ liệu.
- Cài đặt được hàm nhập dữ liệu từ file, xuất dữ liệu hiển thị ra màn hình, và xử lý dữ liệu trong bài toán cụ thể
- Vận dụng được cách đọc, lưu file để giải quyết bài toán cụ thể
- Định nghĩa được biến kiểu cấu trúc trong bài toán cụ thể
- Khai báo được biến kiểu cấu trúc.
- Phân tích được tình huống sử dụng cấu trúc để lưu trữ dữ liệu.
- Cài đặt được hàm nhập, xuất, xử lý dữ liệu bằng cách sử dụng kiểu cấu trúc
- Vận dụng được kiểu cấu trúc để giải quyết bài toán cụ thể

### B. DỤNG CỤ - THIẾT BỊ THÍ NGHIỆM CHO MỘT SV:

STT	Chủng loại – Quy cách vật tư	Số lượng	Đơn vị	Ghi chú
1	– Máy tính (có thể là máy bàn hoặc laptop).	1	cái	

### C. VẬT LIỆU

- Visual C++.Net.
- Phần mềm hỗ trợ dạy học: NetOp School hoặc NetSupport School.

### D. NỘI DUNG THỰC HÀNH

#### I. Tóm tắt lý thuyết

##### 1. Mở - đóng file

- *Mở file:*

```
FILE * fopen(const char * filename, const char * mode);
```

Hàm fopen thực hiện chức năng mở file có tên là filename, dạng mở mode. Nếu mở được thì trả về 1 con trỏ có trị khác NULL, không mở được trả về trị NULL. Tham số mode là một chuỗi kí tự chỉ định dạng mở.

#### Các mode mở file thông dụng:

- “r” : Mở file để đọc (read).

- “w” : Tạo file mới để ghi (write), nếu file này đã tồn tại thì bị ghi đè.
- “a” : Mở để ghi vào cuối file nếu file này đã tồn tại, nếu file này chưa có thì sẽ được tạo mới để ghi (append).
- “r+” : Mở file đã có để cập nhật (cả đọc lẫn ghi).
- “w+” : Mở file mới để được cập nhật (cả đọc lẫn ghi). Nếu file này đã có thì nội dung sẽ bị ghi đè.
- “a+” : Mở để đọc và cập nhật (ghi) nội dung vào cuối file, sẽ tạo mới nếu file này chưa có.

#### Ghi chú:

- Thêm ký tự “t” để mô tả mở file dạng text mode. (Ví dụ : “rt”, “w+t”,...):
- Thêm ký tự “b” để mô tả mở file dạng nhị phân (Ví dụ : “wb”, “a+b”,...).

Ví dụ:

```
FILE *f;
f = fopen("input.txt", "rt");
```

- **Đóng file:**

```
int fclose(<tên con trỏ kiểu file>);
```

Ví dụ:

```
int fclose(FILE * f);
```

Nếu thành công trả giá trị 0, nếu thất bại trả về giá trị EOF ( giá trị -1).

## 2. Đọc file txt kiểu số

```
int fscanf (<tên con trỏ mở file>, "chuỗi định dạng", &bien1, &bien, &bien,... )
```

Ví dụ:

- Khai báo biến số nguyên a, b, c  
int a, b, c;
- Nội dung file txt là:

```
5 6 9
```

- Cách đọc 3 số nguyên lưu vào a,b,c như sau:

```
fscanf(f, "%d", &a);
fscanf(f, "%d", &b);
fscanf(f, "%d", &c);
```

Cách thức gọi hàm:

```
void main()
{
    FILE*f;
    int a, b, c;

    f = fopen("input.txt", "rt");
    fscanf(f, "%d", &a);
    fscanf(f, "%d", &b);
```

```

    fscanf(f, "%d", &c);
    printf("3 gia tri lan luot la: %d, %d, %d", a, b, c);
    getch();
}

```

### 3. Đọc file txt kiểu chuỗi

Cú pháp: `char * fgets (char*s, int n, f);`

Đọc 1 dòng dữ liệu tối đa n byte trong f vào chuỗi s, ngược lại trả về NULL.

Ví dụ: Nội dung file txt là:

Xin chao cac ban!

Cách thức gọi hàm:

```

void main()
{
    FILE*f;
    f = fopen("input.txt", "rt");

    char a[50];
    fgets (a, 50, f);
    printf("\n chuoi la: %s", a);
}

```

### 4. CẤU TRÚC – STRUCT

#### a. Khai báo struct:

##### – Cách 1:

```

struct <Tên cấu trúc>
{
    <Tên kiểu dữ liệu> <Tên thành phần 1>;
    <Tên kiểu dữ liệu> <Tên thành phần 2>;
};

```

##### Ví dụ:

```

struct SinhVien
{
    char mssv[15];
    char hoten[50];
    int solanvang;
};

```

##### – Cách 2:

```

typedef struct <Tên cấu trúc>
{
    <Tên kiểu dữ liệu> <Tên thành phần 1>;
}

```

**<Tên kiểu dữ liệu> <Tên thành phần 2>;  
}<Tên cấu trúc mới>;**

**Ví dụ:**

```
typedef struct SinhVien
{
    char mssv[15];
    char hoten[50];
    int solanvang;
}Student;
```

**b. Khai báo biến sử dụng cấu trúc**

**<Tên cấu trúc> <Tên biến>;**

**Ví dụ:**

```
SinhVien x;
```

Hoặc:

```
Student y;
```

**c. Truy cập thành phần của cấu trúc:**

***Tên\_biến\_cấu\_trúc.tên\_thành\_phần***

**Ví dụ:**

```
SinhVien x;
puts("Nhap ho ten:");
gets(x.hoten);
printf("Nhap so lan vang mat:");
scanf("%d",&x.solanvang);
```

## **II. Bài tập mẫu hướng dẫn thực hành**

**Bài 1.** Cho file txt chứa mảng 1 chiều như sau:

```
5
1 7 9 4 3
```

- **Hãy đọc các giá trị của file txt và lưu vào mảng 1 chiều.**
- **Xuất mảng vừa đọc ra màn hình**

Hướng dẫn

```
#include "stdio.h"
#include "conio.h"
void main()
{
    int n;
    int a[50];

    //Mở file để đọc
    FILE*f;
```

```

f = fopen("input.txt", "rt");
//doc so dau tien trong file luu vao n
fscanf(f, "%d", &n);
//doc n so tiep theo luu vao phan tu cua mang
for(int i=0; i<n; i++)
    fscanf(f, "%d", &a[i]);
//dong file
fclose(f);
//xuat mang ra man hinh
for(int i=0; i<n; i++)
    printf("\t %d", a[i]);
getch();
}

```

## Bài 2. Cho file txt chứa thông tin:

*121*

*Nguyen Thi Thuy*

*23*

- **Đọc giá trị từng dòng dữ liệu của file**
- **Xuất các dữ liệu ra màn hình**

```

#include "stdio.h"
#include "conio.h"
void main()
{
    //Mo file de doc
    FILE*f;
    f = fopen("input.txt", "rt");
    //Doc tung dong du lieu luu vao bien tuong ung
    char ms[15];
    char ten[50];
    char tuoi[10];
    fgets (ms, 50, f);
    fgets (ten, 50, f);
    fgets (tuoi, 50, f);
    //in ket qua ra man hinh
    printf("\n ma so la: %s", ms);
    printf("\n ten la: %s", ten);
    printf("\n tuoi la: %s", tuoi);
    getch();
}

```

**Bài 3.** Khai báo cấu trúc HìnhChuNhat cần thiết lưu trữ thông tin hình chữ nhật, sau đó thực hiện các chức năng: (Lưu ý: Viết chương trình dạng hàm).

***Yêu cầu:***

- Nhập thông tin cho kiểu dữ liệu HìnhChuNhat.
- Xuất thông tin cho kiểu dữ liệu HìnhChuNhat.
- Tính chu vi hình chữ nhật
- Tính diện tích hình chữ nhật

***Hướng dẫn:***

- Khai báo thêm các thư viện cơ bản cho chương trình.

```
#include <conio.h>
```

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

- Khai báo struct HìnhChuNhat cho chương trình.

```
struct HìnhChuNhat
{
    int Dai;
    int Rong;
};
```

- Viết các định nghĩa hàm cần thiết cho chương trình như sau:

```
void NhapThongTinHCN(HìnhChuNhat &pHcn)
```

```
{
    printf("Moi ban nhap chieu dai hcn=");
    scanf_s("%d",&pHcn.Dai);
    printf("Moi ban nhap chieu rong hcn=");
    scanf_s("%d",&pHcn.Rong);
}
```

```
//=====
```

```
void XuatThongTinHCN(HìnhChuNhat pHcn)
```

```
{
    printf("Chieu dai hcn la=%d",pHcn.Dai);
    printf("\nChieu rong hcn la=%d",pHcn.Rong);
}
```

```
//=====
```

```
void TinhChuViHCN(HìnhChuNhat pHcn)
```

```
{
    printf("\nChu vi hcn la=%d", (pHcn.Dai+pHcn.Rong)*2);
}
```

```
//=====
```

```
void TinhDienTichHCN(HìnhChuNhat pHcn)
```

```
{
    printf("\nDien tich hcn la=%d",pHcn.Dai*pHcn.Rong);
}
```

- Viết hàm main để thực thi chương trình.

```
void main()
{
    HìnhChuNhat hìnhchunhat;
    NhapThongTinHCN(hìnhchunhat);
    XuatThongTinHCN(hìnhchunhat);
    TinhChuViHCN(hìnhchunhat);
    TinhDienTichHCN(hìnhchunhat);
    getch();
}
```

### III. Bài tập ở lớp

**Bài 4.** Cho file txt lưu ma trận theo cấu trúc như mẫu sau:

- Dòng đầu tiên là số dòng m của ma trận
- Dòng thứ 2 là số cột n của ma trận
- Các dòng tiếp theo lưu m\*n phần tử của ma trận

Ví dụ:

```
3
4
4 5 8 1
9 -2 8 5
8 5 2 7
```

- Đọc nội dung file và ghi dữ liệu vào biến ma trận
- Tính tổng giá trị chẵn mảng 1 chiều

**Bài 5.** Cho file txt lưu danh sách thí sinh theo cấu trúc như mẫu sau:

- Dòng đầu tiên là số lượng thí sinh N
- 4 dòng tiếp theo lưu thông tin 1 thí sinh gồm (mã số thí sinh, tên thí sinh, năm sinh, quê quán của thí sinh đó)
- Thông tin các thí sinh còn lại lưu trên (N-1)\*4 dòng còn lại

Ví dụ:

```
2
01
Nguyễn văn A
1996
Bến tre
02
Nguyễn Thị N
1987
Hậu giang
```

- Đọc file và xuất nội dung file ra màn hình
- Xuất tên, tuổi của từng thí sinh ra màn hình

**Bài 6.** Khai báo cấu trúc PhanSo cần thiết lưu trữ thông tin phân số, sau đó thực hiện các chức năng:

- Viết hàm nhập vào phân số.
- Viết hàm xuất phân số.
- So sánh phân số với 1 (nhỏ hơn, lớn hơn, bằng).
- Viết hàm tối giản phân số.
- Viết hàm quy đồng 2 phân số.
- Nhập vào 2 phân số. Tính tổng, hiệu, tích và thương của hai phân số. Kết quả phải được tối giản.

**Bài 7.** Khai báo cấu trúc NhanVien lưu trữ thông tin:

- Mã số nhân viên
- Tên nhân viên
- Số ngày công
- Lương 1 ngày công
- Chức vụ

Sau đó thực hiện các chức năng:

- Nhập thông tin cho 1 nhân viên
- Tính lương cho 1 nhân viên
- Tính tiền thưởng cho nhân viên, biết rằng thưởng 50\$ cho nhân viên làm trên 24 ngày.
- Tính tiền phụ cấp chức vụ:
  - Giám đốc: 100\$
  - Phó giám đốc: 80\$
  - Trưởng phòng: 40\$
  - Phó phòng: 20\$
- Tính tiền lương thực lãnh cho nhân viên

**Bài 8.** Khai báo cấu trúc Sinh viên lưu trữ thông tin:

- Mã sinh viên
- Tên sinh viên
- Điểm toán
- Điểm lý
- Điểm hóa
- Sau đó thực hiện các chức năng:
- Đọc thông tin cho 1 sinh viên từ file txt có nội dung sau:

SV2001

Nguyen Van A



7  
8  
3

- Xuất thông tin cho sinh viên
- Tính điểm trung bình cho 1 sinh viên
- Xếp loại cho sinh viên

**Bài 9.** Khai báo cấu trúc Taxi lưu trữ thông tin:

- Mã số
- Tên tài xế
- Địa chỉ đón khách
- Số km đã chạy
- Số chỗ ngồi của xe taxi

Sau đó thực hiện các chức năng:

- Đọc thông tin cho 1 taxi từ file txt có nội dung sau:

2  
Tran Tuan Anh  
140 Le Trong Tan, phuong Tay Thanh, quan Tan Phu, TPHCM  
23  
7

- Xuất thông tin taxi
- Kiểm tra tài xế tên t (nhập từ phím) có chạy taxi đang lưu không?
- Kiểm tra taxi có chạy trên quận/huyện q (người dùng nhập vào) không?
- Viết hàm hiển thị số chỗ ngồi của xe taxi
- Viết hàm tính tiền xe taxi theo bảng giá

Bảng giá	Số chỗ ngồi	
	Đơn giá xe 4 chỗ/km	Đơn giá xe 7 chỗ/km
1-5	12	15
6-10	11	14
>10	9	10

#### IV. Bài tập về nhà

**Bài 10.** Cho file txt lưu danh sách nhân viên theo cấu trúc như mẫu sau:

- Dòng đầu tiên là N lưu số lượng nhân viên
- N dòng tiếp theo lưu thông tin N nhân viên, thông tin của 1 nhân viên được lưu trữ trên 1 dòng gồm có: mã số nhân viên, tên nhân viên, hệ số lương. Mỗi thông tin cách nhau dấu #.

Ví dụ:

2 1# Nguyen văn A #2.67 2#Nguyen Thi N #3.14
--

- Đọc và xuất nội dung file ra màn hình
- Tính lương từng nhân viên, biết rằng, Lương = hệ số \* 1210.

**Bài 11.** Cho file txt lưu danh sách thí sinh theo cấu trúc như mẫu sau:

- Thông tin của 1 sinh viên được lưu trữ trên 1 dòng gồm có: mã số sinh viên, tên sinh viên, năm sinh. Mỗi thông tin cách nhau dấu '- '.

Ví dụ:

1- Nguyen Van A - 1992 2- Nguyen Thi N - 1991
--

- Đọc và xuất nội dung file ra màn hình
- Tính tuổi của mỗi thí sinh (tính đến năm hiện hành).

**Bài 12.** Hãy khai báo một cấu trúc mô tả một điểm trên tọa độ xOy gồm 2 thông tin (tọa độ x, tọa độ y)

Sau đó viết hàm thực hiện các chức năng sau:

- Nhập, xuất 1 điểm.
- Kiểm tra điểm có nằm trên trục tung/trục hoành.
- Tính khoảng cách giữa 2 điểm.
- Tìm điểm đối xứng của 1 điểm qua trục tung/trục hoành.
- Tính khoảng cách từ điểm đang xét đến gốc tọa độ O.

**Bài 13.** Khai báo cấu trúc HoaDon gồm các thông tin:

- Mã hóa đơn
- Tên khách hàng
- Tên sản phẩm
- Giá
- Số lượng

Sau đó viết hàm thực hiện các chức năng sau:

- Nhập thông tin hóa đơn
- Tính thành tiền của hóa đơn (giá \* số lượng)
- Tính giảm giá (giảm 10% cho hóa đơn có số lượng > 50, giảm 5% cho hóa đơn có tên sản phẩm chuot mitsumi)
- Xuất thông tin hóa đơn (Mã hóa đơn, tên hóa đơn, thành tiền, giảm giá).

**Bài 14.** Một điện thoại gồm các thông tin sau:

- Mã điện thoại là chuỗi có 10 ký tự
- Tên điện thoại là chuỗi tối đa 20 ký tự
- Loại điện thoại là chuỗi tối đa 10 ký tự
- Giá bán là một số thực
- Số lượng là một số nguyên

Sau đó viết hàm thực hiện các chức năng sau:

- Xây dựng cấu trúc DIENTHOAI mô tả điện thoại.
- Kiểm tra điện thoại có phải loại GALAXY không ?
- Cho biết điện thoại thuộc phân khúc nào ? biết rằng có 3 phân khúc :  
 + bình dân : giá  $\leq 5$  triệu  
 + trung cấp :  $5 \text{ triệu} < \text{giá} < 10 \text{ triệu}$   
 + cao cấp : giá  $> 10 \text{ triệu}$
- Kiểm tra tình trạng hàng trong kho để thông báo nhập hàng nếu số lượng  $< 5$ .

**--Hết--**