QUẢN LÝ TẬP TIN

# Lý thuyết:

**Các hàm cơ bản về tập tin**

Một hệ thống quản lý tập tin theo chuẩn ANSI bao gồm một số hàm liên quan với nhau. Các hàm thông dụng nhất được liệt kê trong bảng 21.1.

|  |  |
| --- | --- |
| **Name** | **Function** |
| fopen() | Mở một tập tin |
| fclose() | Đóng một tập tin |
| fputc() | Ghi một ký tự vào một tập tin |
| fgetc() | Đọc một ký tự từ một tập tin |
| fread() | Đọc từ một tập tin vào một vùng đệm |
| fwrite() | Ghi từ một vùng đệm vào tập tin |
| fseek() | Tìm một vị trí nào đó trong tập tin |
| fprintf() | Hoạt động giống như printf(), nhưng trên một tập tin |
| fscanf() | Hoạt động giống như scanf(), nhưng trên một tập tin |
| feof() | Trả về true nếu đã đến cuối tập tin (end-of-file) |
| ferror() | Trả về true nếu xảy ra một lỗi |
| rewind() | Đặt lại con trỏ định vị trí (position locator) bên trong tập tin về đầu tập tin |
| remove() | Xóa một tập tin |
| fflush() | Ghi dữ liệu từ một vùng đệm bên trong vào một tập tin xác định |

Bảng 21.1: Các hàm cơ bản về tập tin

Các hàm trên chứa trong tập tin header **stdio.h**

Một con trỏ tập tin (file pointer) rất cần thiết cho việc đọc và ghi các tập tin. Nó là một con trỏ đến một structure chứa thông tin về tập tin. Thông tin bao gồm: tên tập tin, vị trí hiện tại của tập tin, tập tin đang được đọc hay ghi, có bất kỳ lỗi nào xuất hiện hay đã đến cuối tập tin. Người dùng không cần thiết phải biết chi tiết, vì các định nghĩa lấy từ studio.h có bao gồm một khai báo structure tên là **FILE**. Câu lệnh khai báo duy nhất cần thiết cho một con trỏ tập tin là:

FILE \*fp;

Khai báo này cho biết **fp** là một con trỏ trỏ đến một FILE.

**Mở một tập tin văn bản**

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

|  |  |
| --- | --- |
| **Chế độ** | **Ý nghĩa** |
| r | Mở một tập tin văn bản để đọc |
| w | Tạo một tập tin văn bản để ghi |
| a | Nối vào một tập tin văn bản |
| r+ | Mở một tập tin văn bản để đọc/ghi |
| w+ | Tạo một tập tin văn bản để đọc/ghi |
| a+f | Nối hoặc tạo một tập tin văn bản để đọc/ghi |

Nếu phải mở một tập tin **xyz** để ghi, câu lệnh sẽ là:

FILE \*fp;

fp = fopen ("xyz", "w");

Tuy nhiên, một tập tin nói chung được mở bằng cách sử dụng một tập hợp các câu lệnh tương tự như sau:

FILE \*fp;

if ((fp = fopen ("xyz", "w")) == NULL)

{

printf("Cannot open file");

exit (1);

}

**Đóng một tập tin văn bản**

int fclose(FILE \*fp);

Nếu phải đóng một tập tin **xyz** để ghi, câu lệnh sẽ là:

fclose(xyz);

# Bài tập thực hành

**BÀI TẬP 1**

**1. Thời lượng:**

**2. Mô tả bài toán:**

**3. Các bước thực hiện:**

void main()

{

FILE \*f;

char str [80];

// mo file de ghi du lieu

f = fopen("data1", "w");

if (f == NULL)

{

printf ("Cannot open file \n\n");

return(0);

}

printf ("Enter a string: ");

gets (str);

//ghi chuoi vao file

fputs (str, f);

fclose(f);

// mo file de doc du lieu

printf ("\n\n Displaying Contents of File data2\n\n");

if((f=fopen("data1", "r"))==NULL)

{

printf("Cannot open file\n\n");

return(0);

}

{

fgets (str,80,f);

printf ("%s\n\n", str);

}

fclose(f);

}

**BÀI TẬP 2**

**1.Thời lượng :** 30 phút

**2.Mô tả bài toán :**

**3.Các bước thực hiện :**

struct student

{

char hoten[30];

float van;

float toan;

float anh;

};

void main()

{

student a[10];

int n,i;

FILE \*f;

//khai bao con tro file

//ghi file aruct nhi phan

f = fopen("data2.txt", "wb");

if(f==NULL)

{

printf("\nERROR: opening file");

getche();

exit(0);

}

printf("\nNhap so luong sinh vien : ");

scanf("%d", &n);

printf("\nNhap du lieu cua sinh vien: \n");

for(i=0;i<n;i++)

{

printf("\nSinh vien %d :\n", i+1);

fflush stdin ;

printf("\nHo ten %d :", i+1);

gets(a[i].hoten);

printf("Diem van: ");

scanf("%f", &a[i].van);

printf("Diem toan: ");

scanf("%f", &a[i].toan );

printf("Diem anh: ");

scanf("%f", &a[i].anh );

fwrite(&a[i], sizeof(struct student ), 1, f);

}

fclose(f);

printf("\nNhap so luong sinh vien : ");

scanf("%d", &n);

//doc file aruct nhi phan

f = fopen("data2.txt", "rb");

printf("\n\*\*\*\*\*\* audent Details \*\*\*\*\*\*\n");

for(i=0;i<n;i++)

{

fread(&a[i],sizeof(struct student), 1, f);

}

fclose(f);

//in danh sach sinh vien

for(i=0;i<n;i++)

{

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

printf("\n\tHo ten: %s", a[i].hoten);

printf("\n\tDiem van: %0.2f", a[i].van);

printf("\n\tDiem toan: %0.2f", a[i].toan);

printf("\n\tDiem anh: %0.2f", a[i].anh);

}

getche();

}

**BÀI TẬP 3**

**1.Thời lượng :** 30 phút

**2.Mô tả bài toán :**

**3.Các bước thực hiện :**

struct Student

{

char hoten[30];

float van;

float toan;

float anh;

};

void main(){

int n;

Student a[10];

FILE \*f; //khai bao con tro file

//ghi file

f = fopen("data.txt", "w");

if( f==NULL)

{

printf("\nloi! ko mo duoc file");

getche();

return 0;

}

printf("\nNhap so luong sinh vien : ");

scanf("%d", &n);

printf("\nNhap du lieu cua sinh vien: \n");

for(int i=0;i<n;i++)

{

printf("\n\*\*\* Sinh vien %d \*\*\* \n", i+1);

fflush stdin;

printf("\nHo ten %d :", i+1);

gets(a[i].hoten);

printf("Diem van: ");

scanf("%f", &a[i].van);

printf("Diem toan: ");

scanf("%f", &a[i].toan );

printf("Diem anh: ");

scanf("%f", &a[i].anh );

//ghi file

fprintf(f,"%s\n",a[i].hoten);

fprintf(f,"%f %f %f\n",a[i].van,a[i].toan,a[i].anh);

}

fclose(f);

f = fopen("data.txt", "r");

//doc file

printf("\nNhap so luong sinh vien muon doc tu file : ");

scanf("%d", &n);

//doc du lieu tu file

for(int i=0;i<n;i++)

{

fgets(a[i].hoten,50,f);

fscanf(f,"%f %f %f\n",&a[i].van,&a[i].toan,&a[i].anh);

}

fclose(f);

//in mang sinh vien

for(int i=0;i<n;i++)

{

printf("\n\nHo ten :%s\n",a[i].hoten);

printf("\n%f %f %f",a[i].van,a[i].toan,a[i].anh);

}

getche();

}

**BÀI TẬP 4**

**1.Thời lượng :** 30 phút

**2.Mô tả bài toán :**

**3.Các bước thực hiện :**

void main()

{

FILE \*f;

int a[10], n;

//Ghi file van ban , ghi n , phan tu mang

f = fopen("array.txt", "w");

if( f == NULL)

{

printf("Khong the mo file!\n");

exit(0);

}

printf("Nhap so phan tu:");

scanf("%d",&n);

fprintf(f,"%d\n",n);

for (int i = 0; i < n; i++)

{

printf("Nhap vao mot so: ");

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

fprintf(f,"%d ",a[i]);

}

fclose(f);

//Mo file de doc

f = fopen("array.txt", "r");

fscanf(f,"%d\n",&n);

for (int i = 0; i < n; i++)

{

fscanf(f,"%d ",&a[i]);

}

fclose(f);

//in mang

printf("\nMang doc tu file :\n");

for(int i=0;i<n;i++)

printf("%d ",a[i]);

getche();

}

**BÀI TẬP 5**

**1.Thời lượng :** 30 phút

**2.Mô tả bài toán :**

Viết một chương trình C để tính toán kết quả của kỳ thi cuối cùng cho sinh viên.

Người sử dụng tự quyết định số sinh viên cần nhập với số lượng SV được ít hơn 100. Nếu người dùng nhập một số lớn hơn 100 hoặc nhỏ hơn 1, chương trình phải hiển thị thông báo sau:

Người dùng phải nhập các số thích hợp để tiếp tục.

Sau đó, người dùng nhập dữ liệu cho mỗi sinh viên:

1. Nhập tên học sinh.

2. Nhập điểm Math (0,0-10,0).

3. Nhập điểm Physics (0,0-10,0).

4 Nhập điểm Hóa học (0,0-10,0).

5. Nhập điểm Sinh học (0,0-10,0)

Hãy tính toán và hiển thị các điểm trung bình của các đối tượng.

Kết quả của bài thi cuối khoá cho một học sinh được tính như sau:

a. Nếu điểm trung bình là dưới 5.0: FAILED

b. Nếu điểm trung bình là 5-6,5: D Lớp

c. Nếu điểm trung bình là 6,6-8,5: C Grade

d. Nếu điểm trung bình là 8,6-9,5: B Lớp

e. Nếu điểm trung bình lớn hơn 9.5: Một lớp

Sau khi nhập các thông tin chi tiết của tất cả các sinh viên chương trình sẽ hiển thị các chi tiết người dùng nhập vào. Chương trình này cũng nên tính toán và hiển thị Điểm trung bình và quả cho mỗi học sinh. Và ghi dữ liệu vào tập tin "data.txt"

Ví dụ 1:

\*\*\*\*\*\*\*\*\*\*\*\* KẾT QUẢ KỲ THI \*\*\*\*\*\*\*\*\*\*\*\*

Tên của các sinh viên: Nguyễn Văn Sơn

Toán học: 4.3

Vật lý: 3.3

Hóa học: 5.0

Sinh học: 4.8

Điểm trung bình của Nguyễn Văn Sơn là: 4,35

KẾT QUẢ: FAILED

\*\*\*\*\*\*\*\*\*\*\*\* KẾT QUẢ KỲ THI \*\*\*\*\*\*\*\*\*\*\*\*

Tên của các sinh viên: Đoàn Lê Nam

Toán học: 7.3

Vật lý: 8,7

Hóa học: 7.0

Sinh học: 9.8

Điểm trung bình của Đoàn Lê Nam là: 8.20

KẾT QUẢ: C Grade

**3.Các bước thực hiện :**

**BÀI TẬP 2**

**1.Thời lượng :** 30 phút

**2.Mục tiêu:**

**3.Mô tả bài toán :**

**4.Các bước thực hiện :**