



XỬ LÝ FILE TRONG C

Source: <https://viettuts.vn/c-file>

Lợi thế của file

- Là nơi chứa dữ liệu ngay cả sau khi thoát khỏi chương trình.
- Biến và mảng là phương tiện lưu trữ không cố định trong khi tệp là phương tiện lưu trữ vĩnh viễn.
- Rất hữu ích khi làm bài tập mà input đầu vào có quá nhiều dữ liệu, dẫn đến mất rất nhiều thời gian cho việc nhập input này, chưa kể cho những lần nhập sai, khó kiểm soát.

1. Các hàm xử lý file trong C

Có nhiều hàm trong thư viện C để mở, đọc, ghi, tìm kiếm và đóng file. Dưới đây là danh sách các chức năng xử lý file trong C:

#	Hàm	Mô tả
1	fopen()	mở tệp mới hoặc file đang tồn tại
2	fprintf()	ghi dữ liệu vào file
3	fscanf()	đọc dữ liệu từ file
4	fputc()	ghi một ký tự vào file
5	fgetc()	đọc một ký tự từ file
6	fputs()	ghi một chuỗi ký tự vào file
7	fgets()	đọc một chuỗi ký tự từ file
8	fclose()	đóng file
9	fseek()	đặt con trỏ tệp tin vào vị trí đã cho
10	fputw()	ghi một số nguyên vào file
11	fgetw()	đọc một số nguyên từ file
12	ftell()	trả về vị trí hiện tại

#	Hàm	Mô tả
13	rewind()	đặt con trỏ tập tin vào đầu tập tin

Mở file: fopen()

Hàm `fopen()` được sử dụng để mở một file. Cú pháp của hàm `fopen()` được đưa ra dưới đây:

```
FILE *fopen (const char *tên tập tin, chế độ const char *);
```

Bạn có thể sử dụng một trong các chế độ (`mode`) sau trong hàm `fopen()` .

Mode	Mô tả
r	mở file văn bản ở chế độ đọc
w	mở file văn bản ở chế độ ghi
a	mở file văn bản ở chế độ nối thêm nội dung vào nội dung sẵn có trong file.
r+	mở file văn bản ở chế độ đọc và ghi
w+	mở file văn bản ở chế độ đọc và ghi
a+	mở file văn bản ở chế độ đọc và ghi
rb	mở file văn bản ở chế độ đọc và ghi
wb	mở file nhị phân ở chế độ đọc
ab	mở file nhị phân ở chế độ nối thêm nội dung vào nội dung sẵn có trong file.
rb+	mở file nhị phân ở chế độ đọc và ghi
wb+	mở file nhị phân ở chế độ đọc và ghi
ab+	mở file nhị phân ở chế độ đọc và ghi

Đóng file: fclose()

Cú pháp:

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

Hàm fscanf và fprintf()

Hàm fprintf() - Ghi file

Hàm `fprintf()` trong C được sử dụng để ghi các ký tự vào file. Nó gửi dữ liệu được định dạng tới một stream.

Cú pháp:

```
int fprintf(FILE *stream, const char *format [, argument, ...])
```

Ví dụ:

```
#include <stdio.h>

int main() {
    FILE *fp;
    // mở file
    fp = fopen("file.txt", "w");
    // ghi dữ liệu vào file
    fprintf(fp, "Hello C, sử dụng fprintf()...\n");
    // đóng file
    fclose(fp);
    return 0;
}
```

Hàm fscanf() - Đọc file

Hàm `fscanf()` được sử dụng để đọc tập hợp các ký tự từ file. Nó đọc một từ trong file và trả về EOF ở vị trí kết thúc file.

Cú pháp:

```
int fscanf(FILE *stream, const char *format [, argument, ...])
```

Ví dụ 1:

```
#include <stdio.h>

int main() {
    FILE *fp;
    char buff[255];
    fp = fopen("file.txt", "r");
    while (fscanf(fp, "%s", buff) != EOF) {
        printf("%s ", buff);
    }
    fclose(fp);
    return 0;
}
```

Kết quả:

```
Hello C, su dung fprintf()...
```

Ví dụ 2: đọc thông tin sinh viên:

Giả sử chúng ta có file `sinhvien.txt` có nội dung như sau:

Name: Vinh	ID: 1	Marks: 7.5
Name: Phuc	ID: 3	Marks: 9.0
Name: Phi	ID: 2	Marks: 10
Name: Manh	ID: 4	Marks: 9.5
Name: Thang	ID: 5	Marks: 8.0

Sau đây là chương trình đọc file `sinhvien.txt` :

```

#include<stdio.h>

int main() {
    FILE *fp;
    char name[50];
    int id;
    float marks;

    fp = fopen("sinhvien.txt", "r");

    if(fp == NULL) {
        printf("Error opening file\n");
        exit(1);
    }

    printf("Vi du su dung ham fscanf(): \n\n");
    printf("Name:\t\tID\t\tMarks\n");

    // doc thong tin sinh vien
    while( fscanf(fp, "Name: %s\t\tID: %d\t\tMarks: %f\n"
                  , name, &id, &marks) != EOF )
    {
        printf("%s\t\t%d\t\t%.2f\n", name, id ,marks);
    }

    fclose(fp);
    return 0;
}

```

Ví dụ file trong C - **Lưu trữ thông tin nhân viên**

Dưới đây là một ví dụ xử lý tệp để lưu trữ thông tin sinh viên khi người dùng nhập từ console. Chúng ta sẽ lưu trữ id, name và salary của nhân viên.

```
#include <stdio.h>

int main() {
    FILE *fptr;
    int id;
    char name[30];
    int salary;

    fptr = fopen("emp.txt", "w+");

    if (fptr == NULL) {
        printf("File không tồn tại.\n");
        return;
    }

    printf("Nhập id: ");
    scanf("%d", &id);
    fprintf(fptr, "Id = %d\n", id);
    printf("Nhập name: ");
    scanf("%s", name);
    fprintf(fptr, "Name = %s\n", name);
    printf("Nhập salary: ");
    scanf("%f", &salary);
    fprintf(fptr, "Salary = %.2f\n", salary);

    fclose(fptr);
    return 0;
}
```

Kết quả:

```
Nhập id: 1001
Nhập name: David
Nhập salary: 12000
```

File emp.txt

```
Id = 1001
Name = David
Salary = 12000.00
```

Hàm fgets() và fputs() - Đọc và ghi một chuỗi ký tự vào file

Hàm fputs() trong C được sử dụng để ghi một chuỗi ký tự vào file.

Cú pháp:

```
int fputs(const char * s, FILE * stream)
```

Ví dụ:

```
#include <stdio.h>

int main() {
    FILE *fp;
    // open file
    fp = fopen("file2.txt", "w");
    // write string into file
    fputs("hello c programming!", fp);
    // close file
    fclose(fp);
    return 0;
}
```

Hàm fgets() trong C được sử dụng để đọc một dòng ký tự từ một file đã cho.

Cú pháp:

```
char* fgets(char *s, int n, FILE *stream)
```

Ví dụ:

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

int main(){
    FILE *fp;
    char text[300];

    fp=fopen("file2.txt", "r");
    printf("%s",fgets(text,200,fp));

    fclose(fp);
    getch();
    return 0;
}
```

Kết quả:

```
hello c programming!
```

Hàm rewind() trong C - Di chuyển con trỏ về vị trí đầu file

Cú pháp:

```
void rewind(FILE *stream);
```

Ví dụ:

File: myfile.txt

```
Hoc lap trinh C,
```

File: rewind.cpp :

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

int main() {
    FILE *fp;
    char c;

    fp=fopen("myfile.txt","r");

    while ((c = fgetc(fp)) != EOF){
        printf("%c", c);
    }

    // di chuyen con tro file ve vi tri dau tien
    rewind(fp);

    while ((c = fgetc(fp)) != EOF){
        printf("%c", c);
    }

    fclose(fp);
    getch();
    return 0;
}
```

Kết quả:

BÀI TẬP

Hãy giải các bài tập sau (có nhập xuất file đầy đủ)

Bài 1: Máy tính đơn giản (4 điểm)

Tên chương trình: CAL.PAS

Viết chương trình mô phỏng một máy tính đơn giản thực hiện các phép toán số học trên các số nguyên.

Dữ liệu: Nhập từ bàn phím các thông tin theo thứ tự sau: số nguyên **a**, dấu cách, một trong các phép toán (*, +, -, /), dấu cách, số nguyên **b**. Phép "/" là phép chia lấy phần nguyên.

Kết quả: Xuất màn hình một số nguyên – kết quả của việc thực hiện phép toán trên hai số nguyên **a** và **b**.

Lưu ý:

- * Các số nguyên a, b, giá trị của kết quả được cho không vượt quá kích thước lớn nhất của kiểu số nguyên trong Free Pascal.
- * Chương trình phải kiểm tra dữ liệu nhập từ bàn phím.

INPUT	OUTPUT
2 + 3	5

Bài 2: Chuyển đổi đơn vị (4 điểm)

Tên chương trình: CONVERT.PAS

Viết chương trình chuyển đổi đơn vị từ hệ thống đo lường Việt Nam sang hệ thống đo lường Anh-Mỹ (và ngược lại) dựa vào bảng chuyển đổi sau:

Đại lượng	Đơn vị Việt Nam	Đơn vị Anh-Mỹ
Độ dài	1.0000 kilôgam	2.2046 pound
	0.4536 kilôgam	1.0000 pound
Dung tích	1.0000 lít	0.2642 gallon
	3.7854 lít	1.0000 gallon

Lưu ý: Dấu phân cách thập phân là chấm (.)

Dữ liệu: Vào từ file văn bản **CONVERT.INP** gồm 1 dòng chứa số thực, dấu cách và đơn vị đo hệ Việt Nam (hoặc Anh-Mỹ).

Kết quả: Ghi ra file văn bản **CONVERT.OUT** chứa thông tin theo thứ tự: số thực đã được chuyển đổi đơn vị (làm tròn đến 4 chữ số thập phân), dấu cách, đơn vị đo hệ Anh-Mỹ (hoặc Việt Nam) tương ứng.

Đơn vị đo là một trong các kí hiệu **kg** (kilôgam), **lb** (pound), **l** (lít) hoặc **g** (gallon).

CONVERT . INP	CONVERT . OUT
1 kg	2.2046 lb
2 l	0.5284 g
7 lb	3.1752 kg
3.5 g	13.2489 l

Bài 3: Tổng các chữ số (5 điểm)

Tên chương trình: SUMDIGIT.PAS

Cho trước hai số nguyên A và B. Hãy đếm số lượng các số nguyên X thỏa:

- $A \leq X \leq B$
- Tổng các chữ số của X là một số chẵn.

Dữ liệu: Vào từ file văn bản **SUMDIGIT.INP** chứa hai số nguyên A và B ($0 \leq A \leq B \leq 10^9$), cách nhau bởi dấu xuống dòng hoặc dấu cách.

Kết quả: Ghi ra file văn bản **SUMDIGIT.OUT** chứa số lượng số nguyên lớn hơn hoặc bằng A, bé hơn hoặc bằng B và tổng các chữ số là một số chẵn.

SUMDIGIT . INP	SUMDIGIT . OUT
10 20	6

Bài 1: Máy tính đơn giản (4 điểm)**Tên chương trình: CAL.PAS**

INPUT	OUTPUT
2 + 3	5
10 - 25	-15
199 * 5	995
65 / 65	1

Nếu chương trình không kiểm tra dữ liệu nhập (chỉ nhập 1 trong 4 phép toán): trừ 0,5 đ

Bài 2: Chuyển đổi đơn vị (4 điểm)**Tên chương trình: CONVERT.PAS**

CONVERT . INP	CONVERT . OUT
1 kg	2.2046 lb
0.2 lb	0.0907 kg
0.0001 g	0.0004 l
100 l	26.4200 g

Bài 3: Tổng các chữ số (5 điểm)**Tên chương trình: SUMDIGIT.PAS**

SUMDIGIT . INP	SUMDIGIT . OUT
10 20	6
10 10	0
9 99	45
12300 12345	23
9999888 9999999	56