

KỸ THUẬT LẬP TRÌNH C/C++

Dữ liệu kiểu tệp

Thi-Lan Le

Thi-Lan.Le@mica.edu.vn; lan.lethi1@hust.edu.vn

[Webpage: http://www.mica.edu.vn/perso/Le-Thi-Lan](http://www.mica.edu.vn/perso/Le-Thi-Lan)

Nội dung chính

- ◆ Dữ liệu kiểu tệp (file)
- ◆ Các thao tác cơ bản
 - Khai báo
 - Mở tệp (open)
 - Đọc tệp (read)
 - Ghi lên tệp (write)
 - Đóng tệp (close)

Dữ liệu kiểu tệp

◆ Khái niệm kiểu tệp (FILE):

- Tương tự như kiểu chuỗi kí tự, nhưng được lưu trên bộ nhớ ngoài:

$$\mathbf{f = c_1c_2...c_n[EOF]}$$

Trong đó: EOF (End Of File) là kí tự kết thúc tệp

- Lưu ý: kiểu tệp trong C có thể dùng để biểu diễn cả thư mục (directory/folder)

Dữ liệu kiểu tệp

◆ Các tính chất của tệp

- **Tên tệp:** theo quy tắc đặt tên của hệ điều hành
- **Kiểu tệp:** văn bản (text) hoặc nhị phân (binary)
- **Độ dài tệp:** là số kí tự (byte) có trong tệp (không tính kí tự EOF)
- **Vị trí tệp:** là đường dẫn từ thư mục gốc đến thư mục hiện tại chứa tệp
- **Các thuộc tính:** kiểu File hay Directory, read-only hay không, hidden hay không, v.v.

Các thao tác cơ bản

◆ Khai báo biến kiểu tệp:

- Cú pháp: **FILE** *<tên biến>;
- Ý nghĩa: khai báo 1 biến con trỏ kiểu FILE. Trong C luôn dùng loại con trỏ này để thực hiện các thao tác tiếp theo cho tệp.

Các thao tác cơ bản

◆ Mở tệp:

□ Cú pháp:

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

Trong đó:

filename: tên tệp muốn mở, bao gồm cả đường dẫn đến tệp nếu tệp không nằm trong thư mục hiện tại

mode: chế độ mở tệp, bao gồm cả mục đích mở tệp (để đọc, ghi) và loại tệp muốn mở (text hay binary)

Mở tệp

◆ Các mode (chế độ)

Giá trị	Ý nghĩa
“r”	open for reading
“w”	open for writing (file need not exist)
“a”	open for appending (file need not exist)
“r+”	open for reading and writing, start at beginning
“w+”	open for reading and writing (overwrite file)
“a+”	open for reading and writing (append if file exists)
“t”	text file (default)
“b”	binary file

Mở tệp

- ◆ Sau khi một tệp được mở thành công, thì sẽ có một con trỏ vị trí hiện tại trên tệp đó. Con trỏ vị trí này được dùng để đọc và ghi tệp.
- ◆ Vị trí con trỏ này phụ thuộc vào chế độ mở tệp (ta có thể thay đổi nó nhờ hàm `fseek()`):
 - Khi mở tệp để đọc (“r”) hoặc ghi đè (“w”), thì nó nằm ngay vị trí đầu của tệp
 - Khi mở tệp ở chế độ ghi nối đuôi (“a”), thì nó nằm ở vị trí cuối cùng của tệp

Các thao tác cơ bản

◆ Đọc tệp:

- ❑ Đọc từng ký tự: `getc()`, `fgetc()`
- ❑ Đọc chuỗi ký tự: `fgets()`
- ❑ Đọc mảng các phần tử: `fread()`

Program 1: đọc nội dung một tệp sử dụng fgetc()

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

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

```
void printFile(char * fname);
```

```
main()
```

```
{
```

```
    printf("\n---Doc noi dung tep---\n");
```

```
    printFile("QueHuong.txt");
```

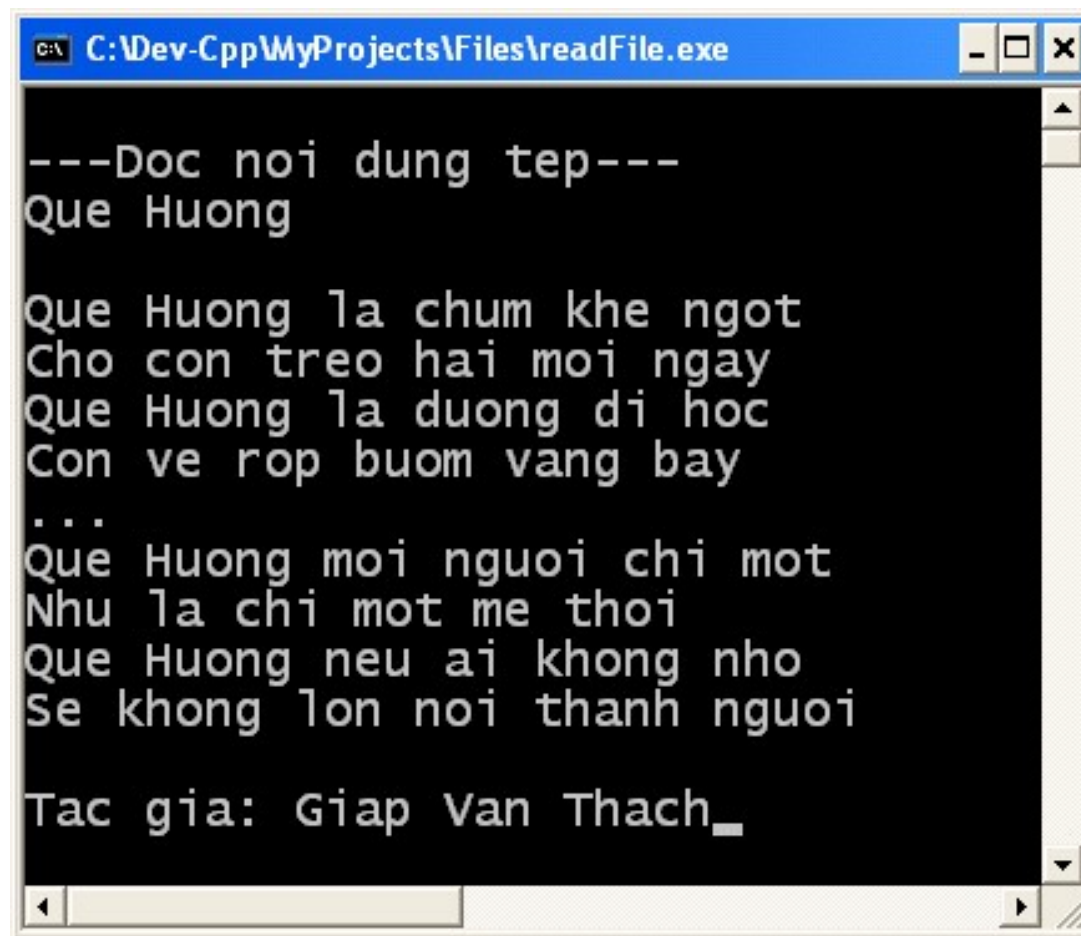
```
    getch();
```

```
}
```

Program 1: đọc nội dung một tệp sử dụng fgetc() (tiếp)

```
void printFile(char * fname)
{
    //open file for reading
    char ch;
    FILE *f=fopen(fname,"r");
    if (f==NULL)
    {
        printf("Error opening file");
        return;
    }
    while ((ch=fgetc(f)) != EOF)
    {
        printf("%c",ch);
    }
    fclose(f);
}
```

Kết quả chạy Program 1



```
C:\Dev-Cpp\MyProjects\Files\readFile.exe

---Doc noi dung tep---
Que Huong

Que Huong la chum khe ngot
Cho con treo hai moi ngay
Que Huong la duong di hoc
Con ve rop buom vang bay
...
Que Huong moi nguoi chi mot
Nhu la chi mot me thoi
Que Huong neu ai khong nho
Se khong lon noi thanh nguoi

Tac gia: Giap Van Thach_
```

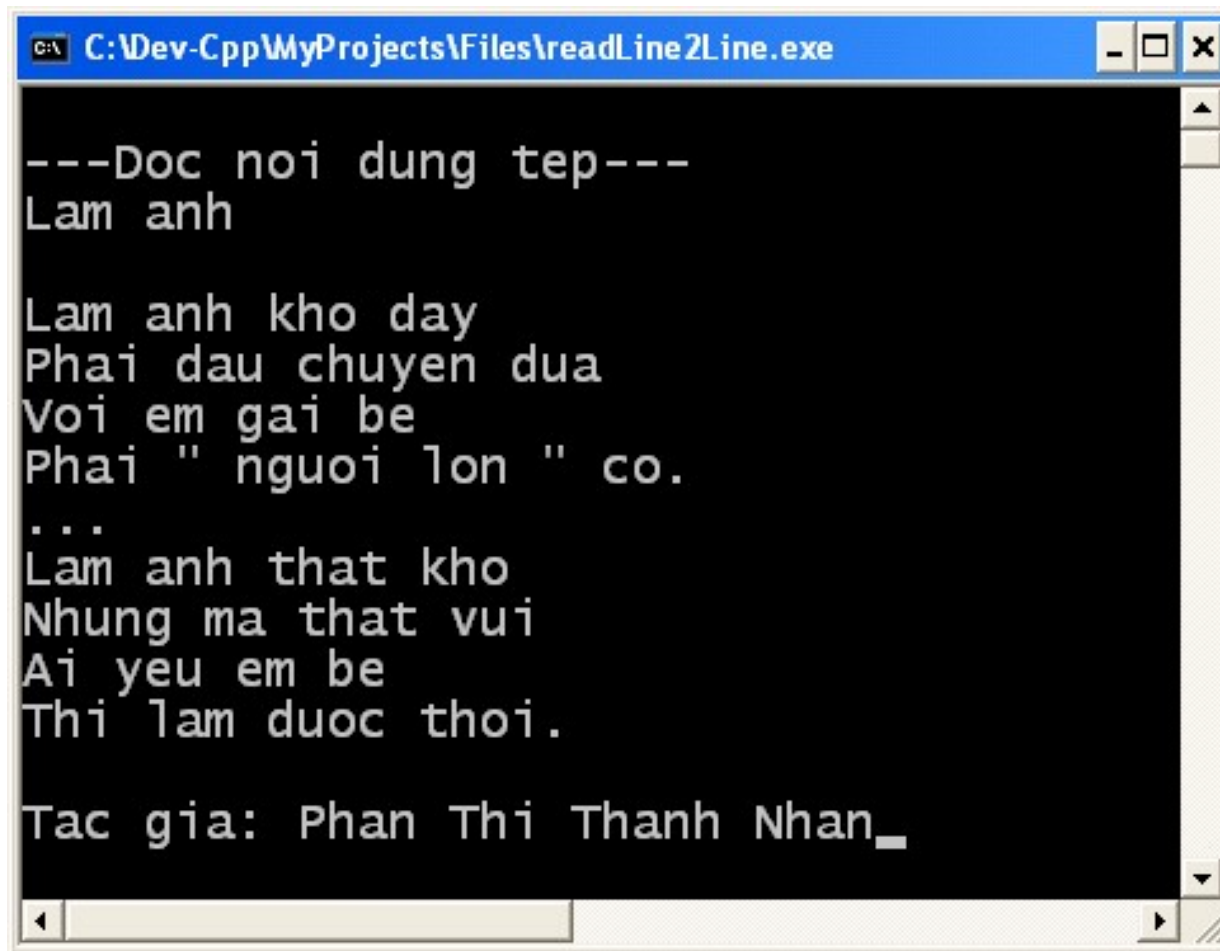
Program 2: đọc nội dung một tệp sử dụng fgets()

```
#include <stdio.h>
#include <conio.h>
#define MAX 300
void printFileByLines(char * fname);
main(){
    printf("\n---Doc noi dung tep---\n");
    printFileByLines("LamAnh.txt");
    getch();
}
```

Program 2: đọc nội dung một tệp sử dụng fgets() (tiếp)

```
void printFileByLines(char * fname) {  
    FILE * pFile;  
    char mystring [MAX];  
    pFile = fopen (fname , "r");  
    if (pFile == NULL){ printf("Error opening file"); return;  
    }  
    while (!feof(pFile)) {  
        if (fgets (mystring , MAX , pFile) !=NULL)  
            printf("%s", mystring);  
    }  
    fclose (pFile);  
}
```

Kết quả chạy Program 2



```
C:\Dev-Cpp\MyProjects\Files\readLine2Line.exe

---Doc noi dung tep---
Lam anh

Lam anh kho day
Phai dau chuyen dua
Voi em gai be
Phai " nguoi lon " co.
...
Lam anh that kho
Nhung ma that vui
Ai yeu em be
Thi lam duoc thoi.

Tac gia: Phan Thi Thanh Nhan_
```

Các thao tác cơ bản

◆ Ghi lên tệp:

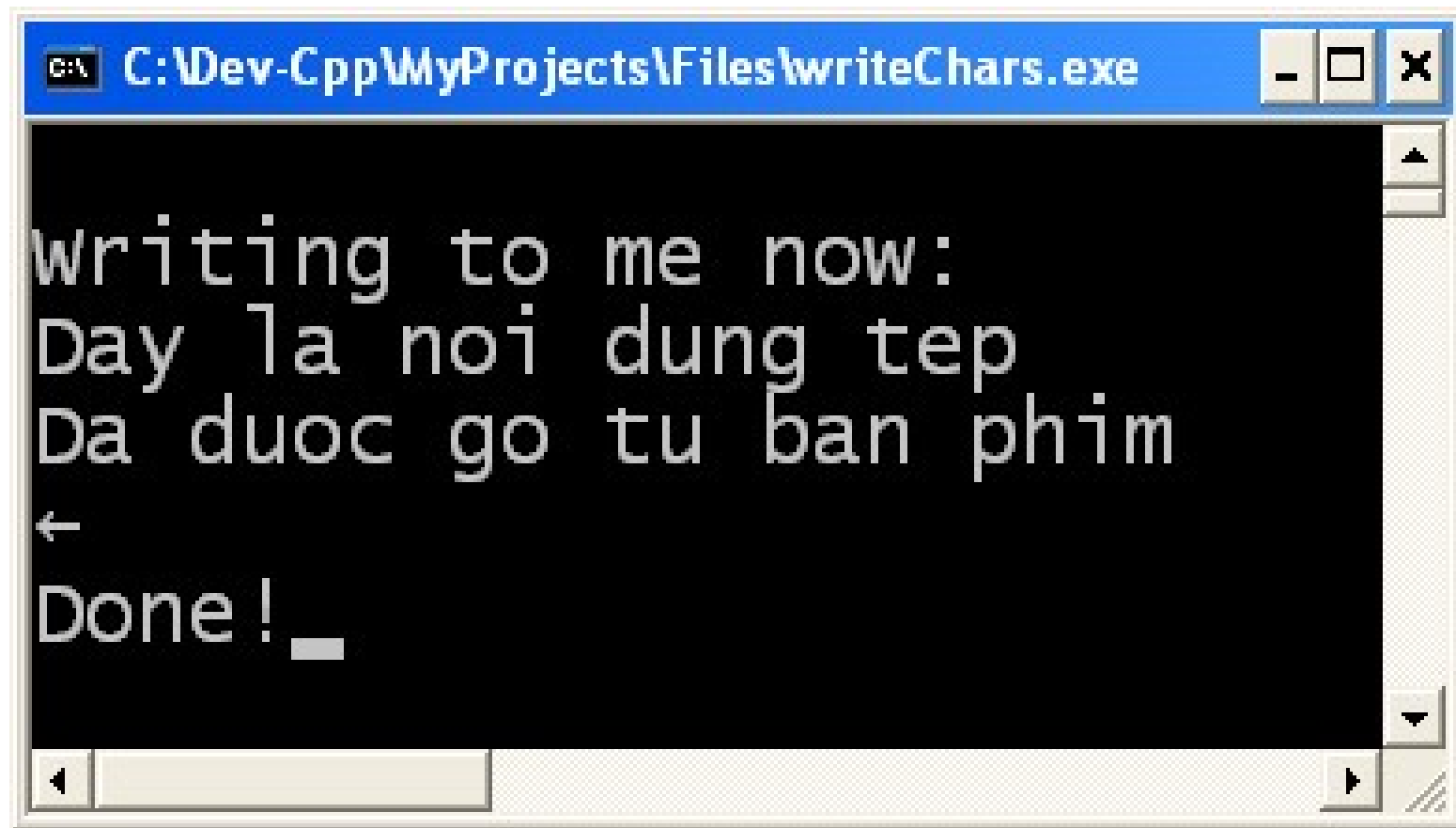
- ❑ Ghi từng ký tự: `putc()`, `fputc()`
- ❑ Ghi chuỗi ký tự: `fputs()`
- ❑ Ghi mảng các phần tử: `fwrite()`

Program 3: ghi nội dung nhập từ bàn phím lên tệp sử dụng các hàm fputc và fputs()

```
#include <stdio.h>
#define END_LINE "\015\012"
#define ESC 27
#define CR 13
void main()
{
    FILE *f1;
    char ch;
    printf("\nWriting to me now:\n");
    //open file for writing
    f1=fopen("data.txt","w");

    if (f1==NULL){
        printf("Error opening file");
        return;
    }
    while ((ch=getche()) != ESC){
        if (ch==CR){
            fputs(END_LINE,f1);
            printf("\n");
        }
        else fputc(ch,f1);
    }
    fclose(f1);
    printf("\nDone!");
} //end main
```

Kết quả chạy Program 3



```
e:\ C:\Dev-Cpp\MyProjects\Files\writeChars.exe
Writing to me now:
Day la noi dung tep
Da duoc go tu ban phim
←
Done!_
```

Các thao tác cơ bản

◆ Truy nhập trực tiếp tệp:

- Hàm fseek(): di chuyển con trỏ vị trí tệp
- Hàm ftell(): trả về vị trí hiện tại của con trỏ vị trí tệp

Truy nhập trực tiếp tệp

◆ Hàm fseek():

- Cú pháp:

int fseek (FILE * stream, int offset, int origin);

- Ý nghĩa: đặt con trỏ vị trí đến vị trí mới thêm một khoảng offset từ vị trí ban đầu origin
- origin có thể là 1 trong 3 hằng số:
 - SEEK_SET: từ đầu tệp
 - SEEK_CUR: từ vị trí hiện tại
 - SEEK_END: từ cuối tệp

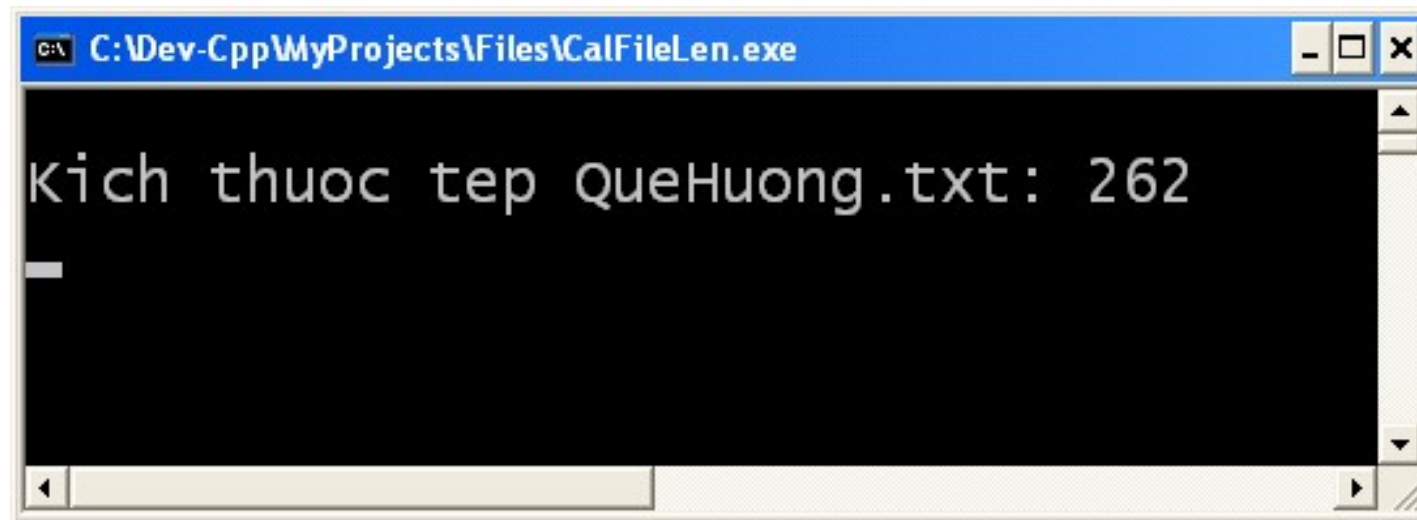
Program 4: Tính kích thước một tệp có tên cho trước

```
#include <stdio.h>
#include <conio.h>
long fileLen(char * fname);
main()
{
    char sFileName[] = "QueHuong.txt";
    printf("\nKich thuoc tep %s: %ld\n",
           sFileName, fileLen(sFileName));
    getch();
}
```

Program 4: Tính kích thước một tệp có tên cho trước (tiếp)

```
long fileLen(char * fname){
    long lfile=0;
    FILE *f=fopen(fname,"r");
    if (f==NULL){
        printf("Error opening file");
        lfile = -1;
    }else {
        fseek(f,0,SEEK_END);
        lfile = ftell(f);
        fclose(f);
    }
    return lfile;
}
```

Kết quả chạy Program 4



A screenshot of a Windows command prompt window. The title bar is blue and contains the text "C:\Dev-Cpp\MyProjects\Files\CalFileLen.exe" along with standard window control buttons (minimize, maximize, close). The main area of the window is black with white text that reads "kich thuoc tep QueHuong.txt: 262". A white cursor is visible on the line below the output. The bottom of the window features a horizontal scrollbar.

```
C:\Dev-Cpp\MyProjects\Files\CalFileLen.exe  
kich thuoc tep QueHuong.txt: 262
```

Các thao tác cơ bản

◆ Đóng tệp:

- Cú pháp: `fclose (FILE* f);`
- Ý nghĩa: kết thúc việc sử dụng file (gồm mở, đọc và ghi). Khi mở tệp để ghi, thì việc đóng tệp sẽ đảm bảo các dữ liệu được ghi sẽ được đẩy ra thiết bị lưu trữ từ vùng bộ đệm,

Tóm tắt nội dung đã học

- ◆ Khái niệm và các tính chất của tệp
- ◆ Các thao tác cơ bản trên tệp

Summary of File operations in C*

Function Name	Operation
fopen()	Creates a new file for use Opens a new existing file
fclose	Closes a file which has been opened for use
getc(), fgetc()	Reads a character from a file
fgets()	Reads a string from a file
putc(), fputc()	Writes a character to a file
fputs()	Writes a string to a file
fprintf()	Writes a set of data values to a file
fscanf()	Reads a set of data values from a file

Summary of File operations in C*

Function Name	Operation
getw()	Reads an integer from a file
putw()	Writes an integer to the file
fread()	Reads an array of elements from a file
fwrite()	Writes an array of elements to a file
fseek()	Sets the position to a desired point in the file
ftell()	Gives the current position in the file
rewind()	Sets the position to the beginning of the file

Bài tập

- Bài 1: viết chương trình đếm số từ có trong một tệp văn bản
- Bài 2: viết chương trình đếm số ký tự có trong một tệp văn bản (không tính các ký tự kết thúc dòng)
- Bài 3: viết chương trình tách đều một tệp làm hai tệp con
- Bài 4: viết chương trình tách một tệp thành các tệp có kích thước không lớn hơn MAX cho trước