

計算機程式設計

C語言 File

郭忠義

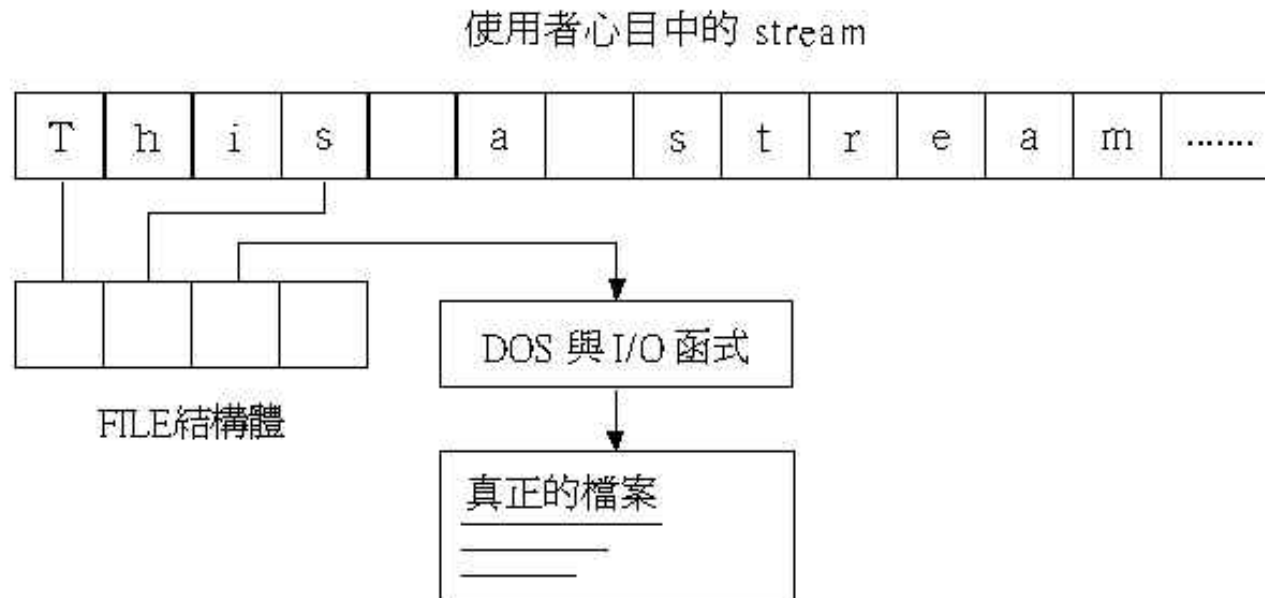
jykuo@ntut.edu.tw

臺北科技大學資訊工程系

檔案處理

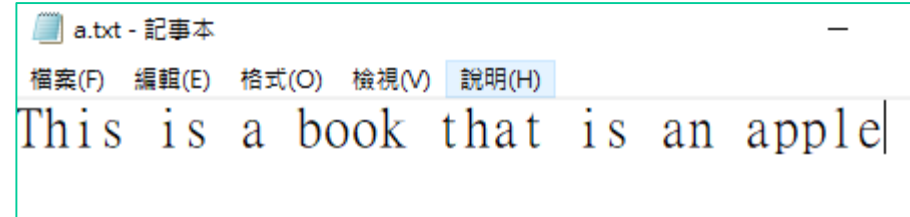
□ FILE 型別

○ Stream是已經open的檔案



檔案開啟關閉與複製

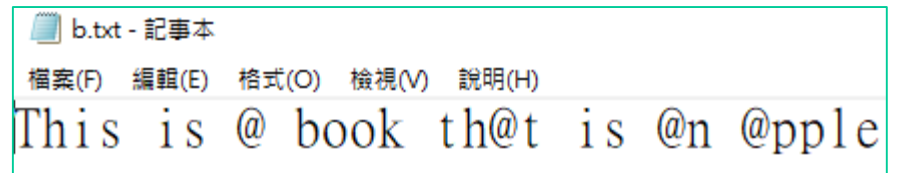
```
#include <stdio.h>
#include <stdlib.h>
void fun(char fileName1[], char fileName2[]) {
    FILE *fp1, *fp2;
    char ch,full[20];
    if ((fp1=fopen(fileName1,"r"))==NULL) {
        printf("cannot open file %s",fileName1);
        exit(0);
    }
    fp2=fopen(fileName2,"w");
    do {
        if ((ch=fgetc(fp1))=='a') ch='@';
        printf("%c",ch);
        fputc(ch,fp2);
    } while (ch!=EOF);
    fclose(fp1);
    fclose(fp2);
}
int main(void) {
    fun("a.txt", "b.txt");
    return 0;
}
```



a.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

This is a book that is an apple



b.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

This is @ book th@t is @n @pple

檔案開啟關閉

❑ FILE *fopen(const char *filename, const char *mode)

- filename – 檔案名稱。

- mode –

- "r" – 讀取，檔案必須存在。
- "w" – 創建空的檔案。若已存，內容被刪除。
- "a" – 附加到檔案中。寫入資料加在檔案末尾。檔案若不存則新創。
- "r+" – 開啟更新檔案讀取和寫入。檔案必須存在。
- "w+" – 創建空檔案，讀取和寫入。
- "a+" – 開啟檔案讀取和追加。

- 回傳值–FILE指標，若失敗將回傳NULL。

❑ int fclose(FILE *stream)

檔案讀取與寫入

❑ int fgetc(FILE *stream)

- stream – 檔案指標，標識要執行的操作串流。
- 回傳值，讀取的字元為unsigned char轉換為int或EOF檔案結束或錯誤。

❑ int fputc(int char, FILE *stream)

- char – 要寫入的字元。
- stream – 檔案指標。
- 回傳值–有錯誤，已寫入相同的字元被回傳。出現錯誤，回傳EOF。

檔案格式化讀取與寫入

- ❑ `int fscanf(FILE *stream, const char *format, ...)`
 - `stream` – 檔案指標。
 - `format` – 與`scanf`用法相同。
- ❑ `int fprintf(FILE *stream, const char *format, ...)`
 - `format` – 與`printf`用法相同。
- ❑ `char *fgets(char *str, int n, FILE *stream)`
 - `n` – 最大被讀取字元數(含結束符號)
 - `str` – 將檔案資料讀到`str`字串
- ❑ `int fputs(const char *str, FILE *stream)`

檔案格式化讀取與寫入

```
#include <stdio.h>
#include <stdlib.h>
void fun(char fileName[]) {
    char s[3][10];
    int year;
    FILE * fp;
    fp = fopen (fileName, "w+");
    fputs("We are in 2022", fp);    //寫入檔案
    rewind(fp); //重回檔案開頭，準備從檔案讀出
    fscanf(fp, "%s %s %s %d", s[0], s[1], s[2], &year);
    printf("%s %s %s %d", s[0], s[1], s[2], year);
    fclose(fp);
}
int main(void) {
    fun("a.txt");
    return 0;
}
```

We are in 2022

檔案格式化讀取與寫入

```
#include <stdio.h>
#include <stdlib.h>
void fun(char fileName[]) {
    char s[80];
    int year;
    FILE * fp;
    fp = fopen (fileName, "w+");
    fprintf(fp, "%s %s in $%d.", "we", "are", 2002); //格式化寫入資料
    rewind(fp);      //重回檔案開頭
    fgets(s, 50, fp); // 從檔案讀出資料
    printf("=> %s", s);
    fclose(fp);
}
int main(void) {
    fun("a.txt");
    return 0;
}
```

=> we are in \$2002.

循序與隨機讀寫

❑ `int fseek(FILE *stream, long int offset, int where)`

- `stream` – 檔案指標。
- `offset` – 從 `where` 開始的位移量。
- `where` – 指定起始點：
 - `SEEK_SET` 檔案開頭
 - `SEEK_CUR` 檔案指標目前位址
 - `SEEK_END` 檔案結尾
 - 成功回傳0，否則回傳非0。

檔案格式化讀取與寫入

```
#include <stdio.h>
#include <stdlib.h>
void fun(char fileName[]) {
    FILE *fp;
    char s[80];
    fp = fopen(fileName,"w+");
    fputs("This is a book", fp);
    fseek( fp, 7, SEEK_SET );
    fputs(" C Programming", fp);
    fseek( fp, 0, SEEK_SET );
    fgets(s, 80, fp);
    printf("=>%s\n", s);
    fclose(fp);
}
int main(void) {
    fun("a.txt");
    return 0;
}
```

=> This is C Programming

Binary檔案 fread

❑ fopen() 設定模式

- `fopen(filename, "rb")`；讀取 binary 檔案
- `fopen(filename, "wb")`；將資料寫入 binary 檔案中
- `fopen(filename, "ab")`；將資料加到 binary 檔案的最後面

❑ `size_t fread(void *p, size_t size, size_t n, FILE *fp)`

- `p` 從檔案讀資料，將資料存入變數位址。
- `size` 從檔案讀資料，每次讀取`size`大小(單位byte)。
- `n` 讀取的個數，每一個都是`size`個byte。
- `fp` 檔案指標。
- 成功回傳讀取的元素總數。若不同於`n`，為發生錯誤或達到檔案末端。

檔案fwrite

```
#include <stdio.h>
#include <stdlib.h>
void fun(char fileName[]) {
    FILE *fp;
    char s2[80], s1[40] = "This is an apple";
    fp = fopen(fileName, "w+");
    fwrite(s1, 2*sizeof(char), 5, fp ); //從 s1 每次寫 2 byte , 寫 5 個/次
    rewind(fp);
    fgets(s2, 60, fp);
    printf("%s\n", s2);
    fclose(fp);
}
int main(void) {
    fun("a.txt");
    return 0;
}
```

This is an

檔案fwrite

- ❑ `size_t fwrite(void *p, size_t size, size_t n, FILE *fp)`
 - `p` 將變數位址資料寫入檔案。
 - `size` 每次從`p`所指位址，讀取`size` byte資料寫入檔案。
 - `n` 寫入的個數，每一個都是`size` byte。
 - `fp` 檔案指標。
 - 成功回傳寫入的元素總數。若不同於`n`，為發生錯誤。

檔案fwrite/fread

```
#include <stdio.h>
#include <string.h>
void fun(char fileName[]) {
    FILE *fp;
    char s1[80], s2[] = "This is an apple";
    long lSize;
    fp = fopen(fileName, "w+");
    fwrite(s2, strlen(s2) + 1, 1, fp);
    fseek(fp, 0, SEEK_END); //檔案指標指到檔案結尾
    lSize = ftell(fp);      //目前檔案指標位置距離
    rewind(fp);             //檔案指標回到檔案開頭
    fread(s1, lSize, 1, fp); //讀檔案
    printf("%s\n", s1);
    fclose(fp);
}
int main() {
    fun("a.txt");
    return(0);
}
```

This is an apple

struct檔案fwrite/fread

```
#include <stdio.h>
#include <string.h>
typedef struct student_s {
    char name[8];
    int id;
    double score;
} student_t;
void fun(char fileName[]) {
    FILE *fp;
    student_t s2, s1 = {"John", 10590011, 90.5};
    long lSize;
    fp = fopen(fileName, "wb+");
    fwrite(&s1, sizeof(student_t), 1, fp);
    fseek(fp, 0, SEEK_END); //檔案指標指到檔案結尾
    lSize = ftell(fp);      //目前檔案指標位置距離
    rewind(fp);             //檔案指標回到檔案開頭
    fread(&s2, sizeof(student_t), 1, fp); //讀檔案
    printf("%s, %d, %.2f\n", s2.name, s2.id, s2.score);
    fclose(fp);
}
int main() { fun("a.bin"); return(0); }
```

John, 10590011, 90.50