

Q: What's the problem ?

```
// 目的：將 data.txt 檔案內的資料，印出來
// 假設 data.txt 的資料為「James」
#include <stdio.h>

int main()
{
    FILE *fptr;
    unsigned char ch;
    int count = 0;

    fptr = fopen ("data.txt", "r");
    if (fptr != NULL)
    {
        while ((ch = getc(fptr)) != EOF)
        {
            count++;
            printf ("Character %d:%c(%x)\n",count, ch, ch);
        }
        fclose (fptr);
    }
    else
        printf ("Can't open file!\n");

    return 0;
}
```

結果：

Character 1:J(4a)

Character 2:a(61)

Character 3:m(6d)

Character 4:e(65)

Character 5:s(73)

Character 6: (ff)

Character 7: (ff)

...

[不斷的印 Character x: (ff)]

不是有判斷當檔案讀取到 EOF(End Of File)時，就停止嗎？為何無窮的一直印下去呢？

Answer:

主要的問題在於程式中的這行程式：

```
while ((ch = getc(fp)) != EOF)
```

`getc()` 傳回一個型別為 `int` 的值，被隱性地轉成 `char` 指定給 `ch`，然後「`(ch = getc(fp))`」整個指定示式的值會被提升到 `int`，以和 `EOF` (End Of File) 常數做比較。`EOF` 在 `stdio.h` 內的定義為《`#define EOF (-1)`》。因為 `ch` 的型態為 `unsigned char`，轉換成 `int` 的值一定不會是負數，這麼一來，迴圈狀況就會一直成立。

解決的方式，較好的方式是將 `ch` 宣告成 `int`，次之是宣告成 `signed char`

- 1) `int ch;`
- 2) `signed char ch;`

Written By James Liang @ 12/4/2006