程式設計
(Programming)

真理大學 資訊工程系 吳汶涓老師

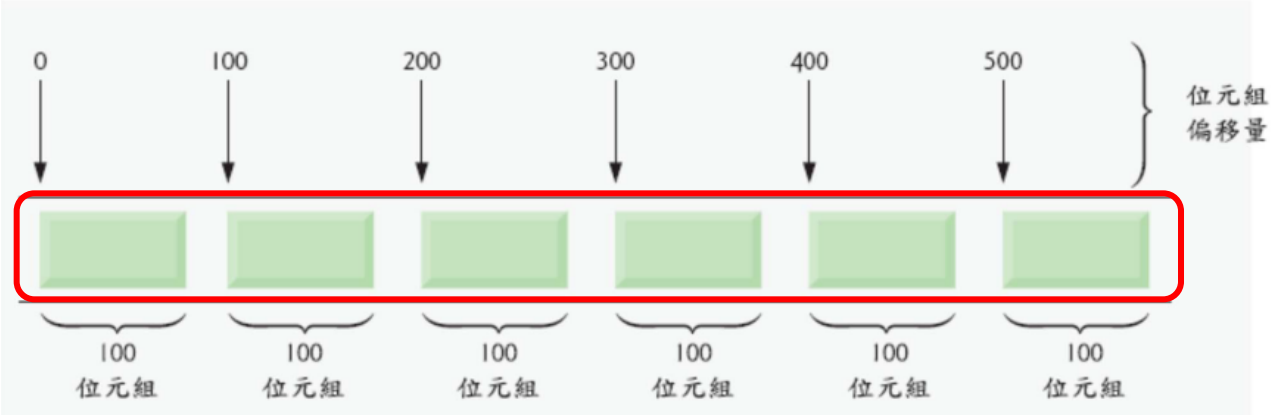
CH11

檔案處理

本章綱要

11.6 隨機存取檔案

- 隨機存取檔案 (Random Access)
  - 存放**沒有先後順序**，但每筆長度一樣空間 (**固定長度紀錄**)
  - 存取個別的紀錄，不需要在紀錄群中進行搜尋 (**快速存取紀錄**)
  - 插入資料不會破壞其他資料
  - 可以更改或刪除舊資料，**不必整個重寫一次**



  - **循序存取** (Sequential Access)：不是固定長度的紀錄
    ↳**想要擷取某個特定記錄，需由關鍵值 (某個)來取得**

11.7 建立隨機存取檔案

- 非格式化I/O函式
  - 儲存結果並非人類可以看得懂的格式
  - **fwrite**：從記憶體傳送指定數目的位元組到檔案中

    之前循序檔是使用 **fprintf(fPtr, "%d", number);**
    現在隨機檔是使用 **fwrite(&number, sizeof(int), 1, fPtr);**

-寫入二進位檔案: 回傳值=fwrite(欲寫入資料, 資料型態大小, 筆數, 檔案指標)

  - **fread**：將指定數目的位元組從檔案傳送到記憶體中

    之前循序檔是使用 **fscanf(fPtr, "%d", &number);**
    現在隨機檔是使用 **fread(&number, sizeof(int), 1, fPtr);**

-讀取二進位檔案: 回傳值=fread(存放資料處, 資料型態大小, 筆數, 檔案指標)

- 練習：

```
FILE *fptr;
int num;
char output[]=”this is a binary file!”;
char input[100];
if((fptr=fopen(“c:\\student.txt”, “wb”))!=NULL){
    num=fwrite(output, sizeof(char), sizeof(output), fptr);
    fclose(fptr);
}
If((fptr=fopen(“c:\\student.txt”, “rb”))!=NULL){
    num=fread(input, sizeof(char), 23, fptr);
    printf(“%s”, input);
    fclose(fptr);
}
return 0;
```

- 寫入struct
  - 檔案處理很少只寫一個欄位到檔案中，故可使用結構

```
fwrite(&blankClient, sizeof(struct clientData), 1, fPtr);
```

  - 範例：製作一個能夠儲存100筆固定長度紀錄的信貸處理系統。每筆紀錄含帳號、姓氏、名字以及餘額。
    - 先宣告結構

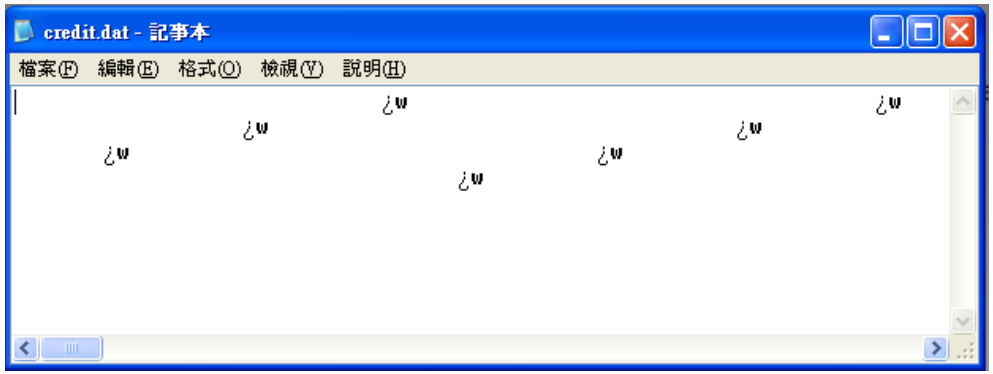
```
6  struct clientData {
7     int acctNum;
8     char lastName[ 15 ];
9     char firstName[ 10 ];
10    double balance;
11 };
```

```
18    struct clientData blankClient = { 0, "", "", 0.0 };
```

```
13 int main( void )
14 {
15    int i;
18    struct clientData blankClient = { 0, "", "", 0.0 };
20    FILE *cfPtr;
23    if ( ( cfPtr = fopen( "credit.dat", "wb" ) ) == NULL ) {
24       printf( "File could not be opened.\n" );
25    }
26    else {
27
29       for ( i = 1; i <= 100; i++ ) {
30          fwrite( &blankClient, sizeof( struct clientData ), 1, cfPtr );
31       }
33       fclose ( cfPtr );
34    }
36    return 0;
38 }
```

fopen 函式會開啟檔案，**wb** 引數表示以二進位模式開啟一個檔案以供寫入

fwrite 將位元組傳輸到隨機存取檔案中



11.8 隨機寫入資料到隨機存取檔案

- fseek
  - 將檔案位置指標設到檔案中某個指定位置

    回傳值 = fseek(檔案指標, offset, 指定起始點);

    - 回傳值為0 (表示移動成功)，大多用於二進位檔案的讀取
    - offset (位移量): 檔案指標指向指定起始點開始的第幾個位元組
    - 指定起始點
      - SEEK_SET (檔頭)
      - SEEK_CUR (目前位置)
      - SEEK_END (檔尾)

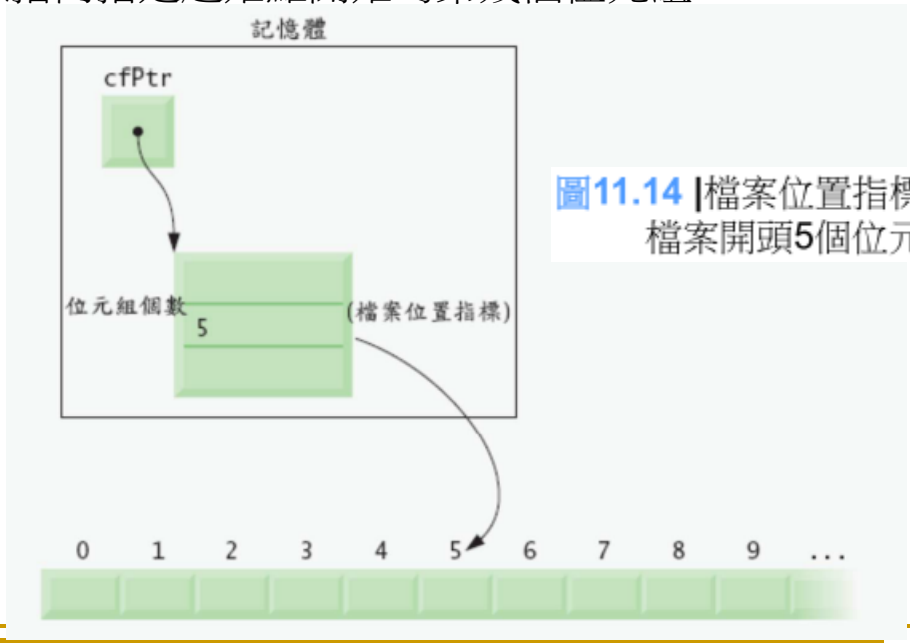
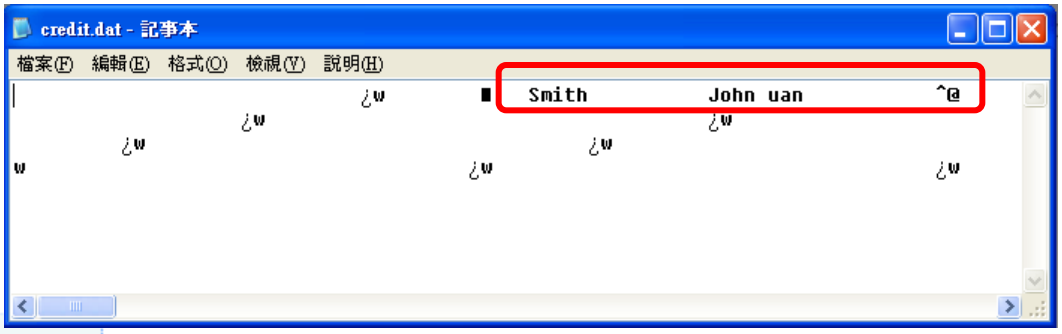


圖11.14 |檔案位置指標指向距離檔案開頭5個位元組的位置

範例：(續)客戶信貸處理系統

```
struct clientData {
   int acctNum;
   char lastName[ 15 ];
   char firstName[ 10 ];
   double balance;
};
```




```
Enter account number ( 1 to 100, 0 to end input )
? 37
Enter lastname, firstname, balance
? Barker Doug 0.00
Enter account number
? 29
Enter lastname, firstname, balance
? Brown Nancy -24.54
Enter account number
? 96
Enter lastname, firstname, balance
? Stone Sam 34.98
Enter account number
? 88
Enter lastname, firstname, balance
? Smith Dave 258.34
Enter account number
? 33
Enter lastname, firstname, balance
? Dunn Stacey 314.33
Enter account number
? 0
```

```
Enter account number (1 to 100, 0 to end input)
?37
Enter lastname, firstname, balance
?Wu WenChuan 100
Enter account number (1 to 100, 0 to end input)
?2
Enter lastname, firstname, balance
?Smith John 120
Enter account number (1 to 100, 0 to end input)
?0
37      Wu      WenChuan        100.000000
請按任意鍵繼續 . . .
```

```
13 int main( void )
14 {
15    FILE *cfPtr;
18    struct clientData client = { 0, "", "", 0.0 };
21    if ( ( cfPtr = fopen( "credit.dat", "wb" ) ) == NULL ) {
22       printf( "File could not be opened.\n" );
23    }
24    else {
27       printf( "Enter account number ( 1 to 100, 0 to end input )\n? " );
29       scanf( "%d", &client.acctNum );
32       while ( client.acctNum != 0 ) {
35          printf( "Enter lastname, firstname, balance\n? " );
38          fscanf( stdin, "%s%s%lf", client.lastName, client.firstName, &client.balance );
39
42          fseek( cfPtr, ( client.acctNum - 1 ) *
43             sizeof( struct clientData ), SEEK_SET );
46          fwrite( &client, sizeof( struct clientData ), 1, cfPtr );
49          printf( "Enter account number\n? " );
50          scanf( "%d", &client.acctNum );
51       }
53       fclose( cfPtr );
54    }
56    return 0;
58 }
```

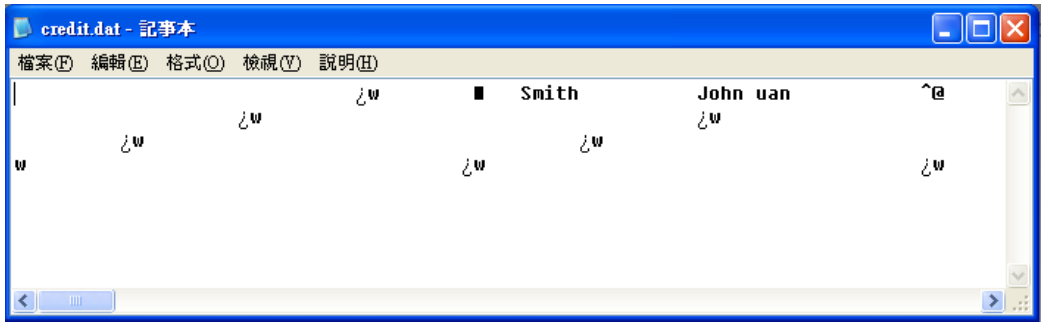
fseek 在隨機存取檔案中搜尋特定的位置

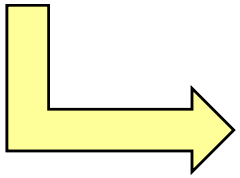
11.9 由隨機存取檔案讀出資料

- fread ：從檔案讀取指定數目的位元組到記憶體

```
fread(&number, sizeof(int), 1, fPtr);
fread(&client, sizeof(struct clientData), 1, fPtr);
```

  - 讀取幾個固定大小的資料





Acct	Last Name	First Name	Balance
29	Brown	Nancy	-24.54
33	Dunn	Stacey	314.33
37	Barker	Doug	0.00
88	Smith	Dave	258.34
96	Stone	Sam	34.98

```c
/* Fig. 11.15: fig11_15.c */
#include <stdio.h>
struct clientData {
   int acctNum;
   char lastName[ 15 ];
   char firstName[ 10 ];
   double balance;
};

int main( void )
{
   FILE *cfPtr;
   struct clientData client = { 0, "", "", 0.0 };
   if ( ( cfPtr = fopen( "credit.dat", "rb" ) ) == NULL ) {
      printf( "File could not be opened.\n" );
   }
   else {
      printf( "%-6s%-16s%-11s%10s\n", "Acct", "Last Name","First Name", "Balance" );
      while ( !feof( cfPtr ) ) {
         fread( &client, sizeof( struct clientData ), 1, cfPtr );

         if ( client.acctNum != 0 ) {
            printf( "%-6d%-16s%-11s%10.2f\n",
               client.acctNum, client.lastName,
               client.firstName, client.balance );
         }
      }
      fclose( cfPtr );
   }
   return 0;
}
```

fread 從隨機存取檔案中讀取位元組到記憶體中

11.10 範例研究：交易處理程式

- 用隨機存取檔案來製作交易處理程式，維護銀行的帳戶資訊 (一開始有100筆空紀錄在credit.dat)

  功能有：

  1. 目前帳戶清單寫入檔案中 (**textFile** 函式)
  2. 更改現有帳戶資料 (**updateRecord** 函式)
  3. 新增新的帳戶 (**newRecord** 函式)
  4. 刪除帳戶 (**deleteRecord** 函式)

```
16 int enterChoice( void );
17 void textFile( FILE *readPtr );
18 void updateRecord( FILE *fPtr );
19 void newRecord( FILE *fPtr );
20 void deleteRecord( FILE *fPtr );
```

```
struct clientData {
   int acctNum;
   char lastName[ 15 ];
   char firstName[ 10 ];
   double balance;
};
```

```
22 int main( void )
23 {
24    FILE *cfPtr;
25    int choice;
28    if ( ( cfPtr = fopen( "credit.dat", "rb+" ) ) == NULL ) {
29       printf( "File could not be opened.\n" );
30    }
31    else {
34       while ( ( choice = enterChoice() ) != 5 ) {
36          switch ( choice ) {
39             case 1:
40                textFile( cfPtr );
41                break;
44             case 2:
45                updateRecord( cfPtr );
46                break;
49             case 3:
50                newRecord( cfPtr );
51                break;
54             case 4:
55                deleteRecord( cfPtr );
56                break;
59             default:
60                printf( "Incorrect choice\n" );
61                break;
63          }
65       }
67       fclose( cfPtr );
68    }
70    return 0;
72 }
```

```
239 int enterChoice( void )
240 {
241   int menuChoice;
244   printf( "\nEnter your choice\n"
245      "1 - store a formatted text file of acounts called\n"
246      "    \"accounts.txt\" for printing\n"
247      "2 - update an account\n"
248      "3 - add a new account\n"
249      "4 - delete an account\n"
250      "5 - end program\n? " );
252   scanf( "%d", &menuChoice );
254   return menuChoice;
256 }
```

- 目前帳戶清單寫入檔案中

```
75 void textFile( FILE *readPtr )
76 {
77    FILE *writePtr;
80    struct clientData client = { 0, "", "", 0.0 };
83    if ( ( writePtr = fopen( "accounts.txt", "w" ) ) == NULL ) {
84       printf( "File could not be opened.\n" );
85    }
86    else {
87       rewind( readPtr );
88       fprintf( writePtr, "%-6s%-16s%-11s%10s\n", "Acct", "Last Name", "First Name","Balance" );
89
92       while ( !feof( readPtr ) ) {
93          fread( &client, sizeof( struct clientData ), 1, readPtr );
96          if ( client.acctNum != 0 ) {
97             fprintf( writePtr, "%-6d%-16s%-11s%10.2f\n",client.acctNum, client.lastName,client.firstName, client.balance );
100         }
102      }
104      fclose( writePtr );
105   }
```

textFile 函式建立一個包含所有帳戶資料的文字檔

- 更改現有帳戶資料

updateRecord 函式改變某個帳戶的餘額

```
110 void updateRecord( FILE *fPtr )
111 {
112    int account;
113    double transaction;
116    struct clientData client = { 0, "", "", 0.0 };
119    printf( "Enter account to update ( 1 - 100 ): " );
120    scanf( "%d", &account );
123    fseek( fPtr, ( account - 1 ) * sizeof( struct clientData ), SEEK_SET );
124
127    fread( &client, sizeof( struct clientData ), 1, fPtr );
130    if ( client.acctNum == 0 ) {
131       printf( "Acount #%d has no information.\n", account );
132    }
133    else {
134       printf( "%-6d%-16s%-11s%10.2f\n\n", client.acctNum, client.lastName, client.firstName, client.balance );
139       printf( "Enter charge ( + ) or payment ( - ): " );
140       scanf( "%lf", &transaction );
141       client.balance += transaction;
143       printf( "%-6d%-16s%-11s%10.2f\n", client.acctNum, client.lastName, client.firstName, client.balance );
148       fseek( fPtr, ( account - 1 ) * sizeof( struct clientData ), SEEK_SET );
152       fwrite( &client, sizeof( struct clientData ), 1, fPtr );
153    }
155 }
```

- 刪除帳戶

```
158  void deleteRecord( FILE *fPtr )
159 {
160
161    struct clientData client;
162    struct clientData blankClient = { 0, "", "", 0 };
163
164    int accountNum;
167    printf( "Enter account number to delete ( 1 - 100 ): " );
168    scanf( "%d", &accountNum );
171    fseek( fPtr, ( accountNum - 1 ) * sizeof( struct clientData ), SEEK_SET );
175    fread( &client, sizeof( struct clientData ), 1, fPtr );
178    if ( client.acctNum == 0 ) {
179       printf( "Account %d does not exist.\n", accountNum );
180    }
181    else {
184       fseek( fPtr, ( accountNum - 1 ) * sizeof( struct clientData ), SEEK_SET );
188       fwrite( &blankClient, sizeof( struct clientData ), 1, fPtr );
190    }
192 }
```

deleteRecord 函式從檔案中移除某個現有的帳戶

- 新增新的帳戶

```
195 void newRecord( FILE *fPtr )
196 {
197
198   struct clientData client = { 0, "", "", 0.0 };
200   int accountNum;
203   printf( "Enter new account number ( 1 - 100 ): " );
204   scanf( "%d", &accountNum );
207   fseek( fPtr, ( accountNum - 1 ) * sizeof( struct clientData ), SEEK_SET );
211   fread( &client, sizeof( struct clientData ), 1, fPtr );
214   if ( client.acctNum != 0 ) {
215      printf( "Account #%d already contains information.\n", client.acctNum );
217   }
218   else {
221      printf( "Enter lastname, firstname, balance\n? " );
222      scanf( "%s%s%lf", &client.lastName, &client.firstName, &client.balance );
225      client.acctNum = accountNum;
228      fseek( fPtr, ( client.acctNum - 1 ) * sizeof( struct clientData ), SEEK_SET );
232      fwrite( &client, sizeof( struct clientData ), 1, fPtr );
234   }
236 }
```

newRecord 函式在檔案中增加一個新的帳戶

補充

- 執行程式時直接下檔名當主程式參數使用

example:
將檔案A內容拷貝
到檔案B中

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

int main(int argc, char *argv[]){
    FILE *fp1, *fp2;
    char ch;
    if(argc!=3){
        printf("argument error \n");
        exit(1);
    }
    fp1=fopen(argv[1],"r");
    fp2=fopen(argv[2],"w");
    while((ch=getc(fp1))!=EOF)
        putc(ch,fp2);
    fclose(fp1);
    fclose(fp2);
    system("pause");
    return 0;
}
```

練習

- 設計一程式可讀取任何文字檔(意味 不能在程式中指定檔名)，將文字內容中所有的英文字母作統計動作，並把統計後之值轉存到另一檔案中。

- 設計一程式有下列兩個功能：
  - 產生n個亂數值並存入一文字檔中
  - 讀入上面文字檔，將數字作分類，奇數放入odd.txt, 偶數放入even.txt檔中。