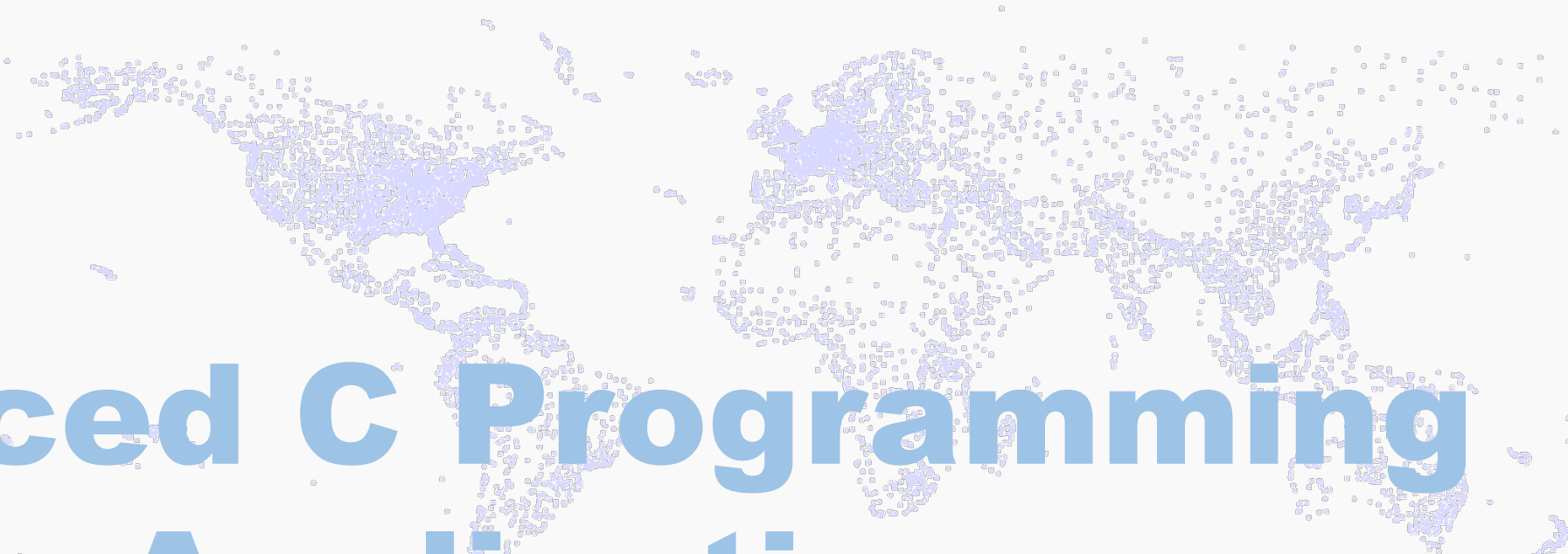


<Adv C & App/>



Advanced C Programming And It's Application

File Processing

Assistant Prof. Chan, Chun-Hsiang

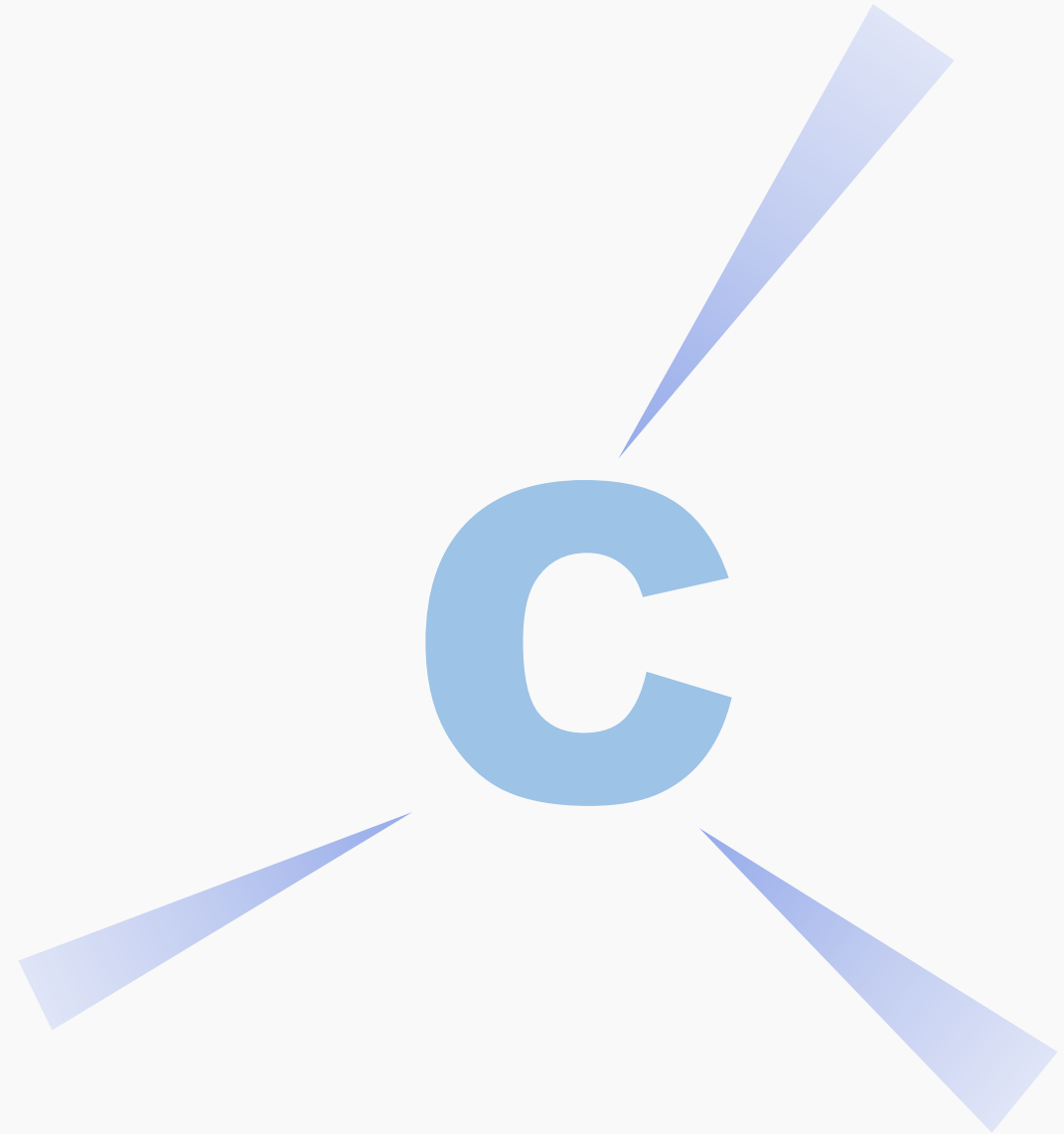
Department of Artificial Intelligence, Tamkang University

Dec. 01, 2021

</ Adv C & App >

大綱

- [1] Data I/O
- [2] fopen
- [3] fprintf & fwrite
- [4] Array to binary/ text file
- [5] fread & fscan
- [6] Read 2D data
- [7] Read csv file
- [8] Assignments
- [9] References



檔案讀寫 Data I/O

正常情況下，我們要做的分析資料不太可能是靠自己一筆一筆的輸入，就算是需要輸入，也不會是打在main()裡面，通常我們都是會用外部資料來源。所以今天的課程，主要是教大家如何讀寫檔案。。。



Request to open



Fetch the data



`fopen`, `fclose`, `fwrite`, `fprintf`, `fscan`, `fread`, ...

fopen

我們在做檔案讀寫的動作時，
第一件事情就是要開啟檔案。

在C語言裡面，通常就是用
fopen配合一個檔案指標開
啟檔案。

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

Mode	Description	Function
r	opens a text file in read mode	Read
w	opens a text file in write mode	Write
a	opens a text file in append mode	Append
r+	opens a text file in read and write mode	Update (w+r)
w+	opens a text file in read and write mode	Empty (w+r)
a+	opens a text file in read and write mode	Read + Append
rb	opens a binary file in read mode	Read
wb	opens a binary file in write mode	Write
ab	opens a binary file in append mode	Append
rb+	opens a binary file in read and write mode	Update (w+r)
wb+	opens a binary file in read and write mode	Empty (w+r)
ab+	opens a binary file in read and write mode	Read + Append

Ref: <https://www.javatpoint.com/file-handling-in-c>

fprintf

(text file)檔案寫入其實就是

- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用fprintf依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

<fprintf/>

fprintf

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

```
int main(){
```

```
    /*Ex 9-1: write text file*/
```

```
    printf("Ex 9-1: write text file\n");
```

```
    int a = 0x31323334, b = 2, c = 100, d = 10;
```

```
    // write into a file
```

```
    FILE *text = fopen("text_9_1.txt", "w");
```

```
    fprintf(text, "%d %d %d %d\n", a, b, c, d);
```

```
    fclose(text);
```

```
}
```

(text file)檔案寫入寫其實就是

- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用fprintf依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

Lab 9-1:

將以下的變數存進一個檔名為lab91_text.txt的text檔案:

```
double a=0, b=2.3, c=3.14;
```

6

fwrite

(bin file)檔案寫入其實就是

- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用fwrite依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

<fwrite/>

fwrite

(**bin file**)檔案寫入寫其實就是

- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用**fwrite**依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

Lab 9-2:

將以下的變數存進一個檔名為lab91_text.sav的bin檔案:

```
double a=0, b=2.3, c=3.14;
```

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

```
int main(){
```

```
    /*Ex 9-2: write bin file*/
```

```
    printf("Ex 9-2: write bin file\n");
```

```
    int a = 0x31323334, b = 2, c = 100, d = 10;
```

```
    // write into a file
```

```
    FILE *bin = fopen("text_9_2.sav", "wb");
```

```
    fwrite(&a, sizeof(a), 1, bin);
```

```
    fwrite(&b, sizeof(b), 1, bin);
```

```
    fwrite(&c, sizeof(c), 1, bin);
```

```
    fwrite(&d, sizeof(d), 1, bin);
```

```
    fclose(bin);
```

```
}
```


- (bin file)檔案寫入寫其實就是
- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用fwrite依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

檔案寫入 Array2TextFile

我們剛剛數將變數寫入檔案，但如果今天我們要寫入的是矩陣，那應該怎麼做呢？

```
#include <stdio.h>
int main(){
    /*Ex 9-3: write an array into a text file*/
    printf("Ex 9-3: write an array into a text file\n");
    int a[20] = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19};
    int i;
    // write into a file
    FILE *text = fopen("text_9_3.txt", "w");
    for (i=0; i<sizeof(a)/sizeof(a[0]); i++){
        fprintf(text, "%d\t", a[i]);
    }
    fclose(text);
}
```

Lab 9-3:
寫一個function可以自動產生Fibonacci number，將前10項存進一個矩陣中，並寫入一個檔名為lab93_text.txt的text檔案中。

- (bin file)檔案寫入寫其實就是
- (1) 利用一個pointer去指向檔案的位置
- (2) 利用fopen打開檔案
- (3) 利用fwrite依照需要的格式印到檔案
- (4) 利用fclose把檔案關掉

檔案寫入 Array2BinFile

同樣的道理，如果我們今天要改成寫入二進位的檔案裏面時...

```
#include <stdio.h>
int main(){
    /*Ex 9-4: write an array into a binary file*/
    printf("Ex 9-4: write an array into a binary file\n");
    int a[20] = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19};
    int i;
    // write into a file
    FILE *bin = fopen("text_9_4.sav", "wb");
    for (i=0; i<sizeof(a)/sizeof(a[0]); i++){
        fwrite(&a[i], sizeof(a[i]), 1, bin);
    }
    fclose(bin);
}
```

Lab 9-4:
寫一個function可以自動產生Fibonacci number，將前10項存進一個矩陣中，並寫入一個檔名為lab94_text.sav的bin檔案中。

Read a file

既然我們能寫入，讀取檔案的改念其實也很相近...

先前我們是講說利用**fprintf**將變數印到檔案中；同理，現在我們要做的事情就是用**fread**將data讀出來，並存進所對應的變數之中。

大致的流程：

(1) **fopen**開啟檔案

(2) **fread (for bin)/fscanf (for txt)**讀取資料

(3) **fclose**關閉檔案

fread

```
#include <stdio.h>
int main(){
    /*Ex 9-5: read a binary file*/
    printf("Ex 9-5: read a binary file\n");
    int a[20] = {0};
    int i;
    // read a file
    FILE *bin = fopen("text_9_4.sav", "rb");
    for (i=0;i<sizeof(a)/sizeof(a[0]);i++){
        fread(&a[i], sizeof(a[i]), 1, bin);
    }
    fclose(bin);
}
```

```
// print array element
for (i=0;i<sizeof(a)/sizeof(a[0]);i++){
    printf("%d\t",a[i]);
    if ((i+1)%5==0){
        putchar('\n');
    }
}
```

大致的流程:

(1) `fopen` 開啟檔案

(2) `fread` (for bin)/`fscanf` (for txt) 讀取資料

(3) `fclose` 關閉檔案

Lab 9-5:

將 lab94_text.sav 的 bin 檔案中資料讀取回來存在一個矩陣，並列印出來!

大致的流程:
(1) `fopen`開啟檔案
(2) `fread` (for bin)/`fscanf` (for txt)讀取資料
(3) `fclose`關閉檔案

fscan

```
#include <stdio.h>
int main(){
    /*Ex 9-6: read a text file*/
    printf("Ex 9-6: read a text file\n");
    int a[20] = {0};
    int i;
    // read a file
    FILE *text = fopen("text_9_3.txt", "r");
    for (i=0;i<sizeof(a)/sizeof(a[0]);i++){
        fscanf(text, "%d", &a[i]);
    }
    fclose(text);
}
```

```
// print array element
for (i=0;i<sizeof(a)/sizeof(a[0]);i++){
    printf("%d\t",a[i]);
    if ((i+1)%5==0){
        putchar('\n');
    }
}
```

Lab 9-6:
將lab93_text.txt的text檔案中資料讀取回來存在一個矩陣，並列印出來!

<read2Ddata/>

讀取二維資料

正常情況下，我們會遇到的資料往往不是一維矩陣，至少都是二維矩陣或是以上(such as image)，那麼如何匯入一個二維矩陣資料呢？

首先，我們可以先宣告一個 **2D zero array**，再填數字進去即可！說到這裡大家有想法了嗎？還是會想到你可能會遇到甚麼問題？

其實會有很多問題發生 ...

- (1) 標頭不要怎麼辦？
- (2) 分隔符號不是空格或是tab，該怎麼辦？
- (3) 一定要是txt檔嗎？那csv可以嗎？

<read2Ddata/>

讀取二維資料

既然可以讀一維陣列資料，一定也可以讀二維矩陣。

我們這邊練習一個情境，我們將新北市人口資料複製貼上到記事本或是sublime裡面，另存為pop_ntc.txt。

Url: <https://www.ca.ntpc.gov.tw/home.jsp?id=88f142fb0f4a0762>

人口統計(110年10月)列表						
隸屬區	里數	鄰數	戶數	男	女	合計
板橋區	126	2,515	219,474	269,207	284,577	553,784
三重區	119	2,564	155,491	187,529	194,797	382,326
中和區	93	2,998	169,892	196,965	210,399	407,364
永和區	62	1,249	91,708	102,563	113,921	216,484
新莊區	84	1,842	162,638	205,512	216,457	421,969
新店區	69	1,482	130,834	144,272	156,609	300,881
土城區	47	1,236	90,448	116,819	120,795	237,614
蘆洲區	38	727	73,954	98,750	102,975	201,725
樹林區	42	1,039	67,810	90,837	91,474	182,311

```

1 板橋區 126 2515 219474 269207 284577 553784
2 三重區 119 2564 155491 187529 194797 382326
3 中和區 93 2998 169892 196965 210399 407364
4 永和區 62 1249 91708 102563 113921 216484
5 新莊區 84 1842 162638 205512 216457 421969
6 新店區 69 1482 130834 144272 156609 300881
7 土城區 47 1236 90448 116819 120795 237614
8 蘆洲區 38 727 73954 98750 102975 201725
9 樹林區 42 1039 67810 90837 91474 182311
10 鶯歌區 20 434 32582 44114 44028 88142
11 三峽區 28 608 44798 58042 58123 116165
12 淡水區 42 698 84774 88496 96488 184984
13 汐止區 50 1141 92166 100174 105691 205865
14 瑞芳區 34 426 16121 19439 19147 38586
15 五股區 20 515 35582 45351 45065 90416
16 泰山區 17 466 29014 38251 39352 77603
17 林口區 17 443 50513 60308 63826 124134
18 八里區 10 195 16014 19962 20118 40080
19 深坑區 8 239 9866 11824 11854 23678
20 石碇區 12 107 3392 4102 3405 7507

```

總共應該要有29筆資料

</read2Ddata>

<read2Ddata/>

讀取二維資料

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

```
int main(){
```

```
/*Ex 9-7: read a 2D text file*/
```

```
printf("Ex 9-7: read a 2D text file\n");
```

```
int i, j, ROW=29, COL=6;
```

string

integer

```
int a[ROW][COL];
```

```
char s[ROW][10];
```

1	板橋區	126	2515	219474	269207	284577	553784
2	三重區	119	2564	155491	187529	194797	382326
3	中和區	93	2998	169892	196965	210399	407364
4	永和區	62	1249	91708	102563	113921	216484

```
// read a file
```

```
FILE *text = fopen("pop_ntc.txt", "r");
```

```
for (i=0;i<ROW;i++){
```

```
for (j=0;j<COL+1;j++){
```

```
if (j==0){
```

```
fscanf(text, "%s", &s[i][j]);
```

```
}else{
```

```
fscanf(text, "%d", &a[i][j-1]);
```

```
}
```

```
}
```

```
}
```

```
fclose(text);
```

</read2Ddata>

2D array for int type

2D array for char type

Integer array

String



End sym.

讀取二維資料

	string	integer					
1	板橋區	126	2515	219474	269207	284577	553784
2	三重區	119	2564	155491	187529	194797	382326
3	中和區	93	2998	169892	196965	210399	407364
4	永和區	62	1249	91708	102563	113921	216484

// print array element

```
for (i=0;i<ROW;i++){
```

```
printf("%s\t",s[i]);
```

```
for (j=0;j<COL;j++){
```

```
printf("%d\t",a[i][j]);
```

```
}
```

```
putchar('\n');
```

```
}
```

```
}
```

Integer array

String



End sym.

<read2Ddata/>

讀取二維資料

Lab 9-7:

利用Ex 9-7的概念，將戶籍相關資料與性別資料分開儲存在兩個矩陣之中：

```
int houseData[ROW][3];
```

// 里數、鄰數、戶數

```
int sexData[ROW][2]; // 男、女
```

```
char district[ROW][10]; // 隸屬區
```

最後再格式化印出來，如右圖。

```
Lab 9-7: read a 2D text file
板橋區      126    2515    219474    269207    284577
三重區      119    2564    155491    187529    194797
中和區       93    2998    169892    196965    210399
永和區       62    1249     91708    102563    113921
新莊區       84    1842    162638    205512    216457
新店區       69    1482    130834    144272    156609
土城區       47    1236     90448    116819    120795
蘆洲區       38     727    73954     98750    102975
樹林區       42    1039    67810     90837     91474
鶯歌區       20     434    32582     44114     44028
三峽區       28     608    44798     58042     58123
淡水區       42     698    84774     88496     96488
汐止區       50    1141    92166    100174    105691
瑞芳區       34     426    16121     19439     19147
五股區       20     515    35582     45351     45065
泰山區       17     466    29014     38251     39352
林口區       17     443    50513     60308     63826
八里區       10     195    16014     19962     20118
深坑區        8     239     9866    11824    11854
石碇區       12     107     3392     4102     3405
坪林區        7      78     2566     3608     3038
三芝區       13     254     9638    11502    10905
石門區        9     124     4223     5847     5378
金山區       15     201     7224    10241    10620
萬里區       10     186     7550    10772    10773
平溪區       12     120     2223     2433     1911
雙溪區       12     244     3684     4490     3773
貢寮區       11     202     4333     5914     5674
烏來區        5      43     1916     3118     3254
```

</read2Ddata>

讀取csv檔案

一般來說最常見的open data檔案格式，就是csv或是json檔。我們這邊主要先來介紹csv檔，csv檔通常是以逗號做分隔，也有部分檔案是以空白或是tab做分隔。除此之外，一般正常的資料檔案都會有標頭，那我們又要如何將標頭存取下來呢？

- Issue 1: read a csv file
- Issue 2: other separators (such as comma)
- Issue 3: header or useless metadata

Download CSV File

我們用剛剛的新北市人口資料做為範例，選擇上方的下載csv檔。
先利用MS Excel開啟，再用記事本/sublime開一次。

	A	B	C	D	E	F	G	H	I
1	年	月	隸屬區	里數	鄰數	戶數	男	女	合計
2	110	10	板橋區	126	2515	219474	269207	284577	553784
3	110	10	三重區	119	2564	155491	187529	194797	382326
4	110	10	中和區	93	2998	169892	196965	210399	407364
5	110	10	永和區	62	1249	91708	102563	113921	216484
6	110	10	新莊區	84	1842	162638	205512	216457	421969
7	110	10	新店區	69	1482	130834	144272	156609	300881
8	110	10	土城區	47	1236	90448	116819	120795	237614
9	110	10	蘆洲區	38	727	73954	98750	102975	201725
10	110	10	樹林區	42	1039	67810	90837	91474	182311
11	110	10	鶯歌區	20	434	32582	44114	44028	88142
12	110	10	三峽區	28	608	44798	58042	58123	116165
13	110	10	淡水區	42	698	84774	88496	96488	184984
14	110	10	汐止區	50	1141	92166	100174	105691	205865
15	110	10	瑞芳區	34	426	16121	19439	19147	38586
16	110	10	五股區	20	515	35582	45351	45065	90416
17	110	10	泰山區	17	466	29014	38251	39352	77603
18	110	10	林口區	17	443	50513	60308	63826	124134
19	110	10	八里區	10	195	16014	19962	20118	40080
20	110	10	深坑區	8	239	9866	11824	11854	23678
21	110	10	石碇區	12	107	3392	4102	3405	7507
22	110	10	坪林區	7	78	2566	3608	3038	6646
23	110	10	三芝區	13	254	9638	11502	10905	22407
24	110	10	石門區	9	124	4223	5847	5378	11225
25	110	10	金山區	15	201	7224	10241	10620	20861
26	110	10	萬里區	10	186	7550	10772	10773	21545
27	110	10	平溪區	12	120	2223	2433	1911	4344
28	110	10	雙溪區	12	244	3684	4490	3773	8263
29	110	10	貢寮區	11	202	4333	5914	5674	11588
30	110	10	烏來區	5	43	1916	3118	3254	6372
31	總計			1032	22376	1620428	1960442	2054427	4014869

```

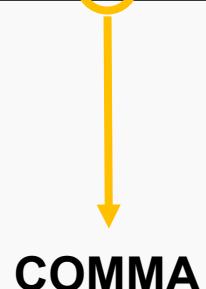
1 年,月,隸屬區,里數,鄰數,戶數,男,女,合計
2 110,10,板橋區,126,2515,219474,269207,284577,553784
3 110,10,三重區,119,2564,155491,187529,194797,382326
4 110,10,中和區,93,2998,169892,196965,210399,407364
5 110,10,永和區,62,1249,91708,102563,113921,216484
6 110,10,新莊區,84,1842,162638,205512,216457,421969
7 110,10,新店區,69,1482,130834,144272,156609,300881
8 110,10,土城區,47,1236,90448,116819,120795,237614
9 110,10,蘆洲區,38,727,73954,98750,102975,201725
10 110,10,樹林區,42,1039,67810,90837,91474,182311
11 110,10,鶯歌區,20,434,32582,44114,44028,88142
12 110,10,三峽區,28,608,44798,58042,58123,116165
13 110,10,淡水區,42,698,84774,88496,96488,184984
14 110,10,汐止區,50,1141,92166,100174,105691,205865
15 110,10,瑞芳區,34,426,16121,19439,19147,38586
16 110,10,五股區,20,515,35582,45351,45065,90416
17 110,10,泰山區,17,466,29014,38251,39352,77603
18 110,10,林口區,17,443,50513,60308,63826,124134
19 110,10,八里區,10,195,16014,19962,20118,40080
20 110,10,深坑區,8,239,9866,11824,11854,23678
21 110,10,石碇區,12,107,3392,4102,3405,7507
22 110,10,坪林區,7,78,2566,3608,3038,6646
23 110,10,三芝區,13,254,9638,11502,10905,22407
24 110,10,石門區,9,124,4223,5847,5378,11225
25 110,10,金山區,15,201,7224,10241,10620,20861
26 110,10,萬里區,10,186,7550,10772,10773,21545
27 110,10,平溪區,12,120,2223,2433,1911,4344
28 110,10,雙溪區,12,244,3684,4490,3773,8263
29 110,10,貢寮區,11,202,4333,5914,5674,11588
30 110,10,烏來區,5,43,1916,3118,3254,6372
31 總計,,1032,22376,1620428,1960442,2054427,4014869

```

```

1 年,月,隸屬區,里數,鄰數,戶數,男,女,合計
2 110,10,板橋區,126,2515,219474,269207,284577,553784
3 110,10,三重區,119,2564,155491,187529,194797,382326

```



Preprocessing CSV File

為了方便理解如何讀取CSV檔，我們需要做三件事情：

- (1) 將行政區的名稱改成英文 (可以Google查一下!)
- (2) 將Header的中文名稱改為英文 → 另存新檔為popWithHeader.csv
- (3) 將Header去掉 → 另存新檔為pop.csv

pop.csv

```

1 110,10,Banqiao,126,2515,219474,269207,284577,553784
2 110,10,Sanchong,119,2564,155491,187529,194797,382326
3 110,10,Zhonghe,93,2998,169892,196965,210399,407364
4 110,10,Yonghe,62,1249,91708,102563,113921,216484
5 110,10,Xinzhuang,84,1842,162638,205512,216457,421969
6 110,10,Xindian,69,1482,130834,144272,156609,300881
7 110,10,Tucheng,47,1236,90448,116819,120795,237614
8 110,10,Luzhou,38,727,73954,98750,102975,201725
9 110,10,Shulin,42,1039,67810,90837,91474,182311
10 110,10,Yingge,20,434,32582,44114,44028,88142
11 110,10,Sanxia,28,608,44798,58042,58123,116165
12 110,10,Tamsui,42,698,84774,88496,96488,184984
13 110,10,Xizhi,50,1141,92166,100174,105691,205865
14 110,10,Ruifang,34,426,16121,19439,19147,38586
15 110,10,Wugu,20,515,35582,45351,45065,90416
16 110,10,Taishan,17,466,29014,38251,39352,77603

```

popWithHeader.csv

```

1 year,month,district,village,neighborhood,house,men,women,total population
2 110,10,Banqiao,126,2515,219474,269207,284577,553784
3 110,10,Sanchong,119,2564,155491,187529,194797,382326
4 110,10,Zhonghe,93,2998,169892,196965,210399,407364
5 110,10,Yonghe,62,1249,91708,102563,113921,216484
6 110,10,Xinzhuang,84,1842,162638,205512,216457,421969
7 110,10,Xindian,69,1482,130834,144272,156609,300881
8 110,10,Tucheng,47,1236,90448,116819,120795,237614
9 110,10,Luzhou,38,727,73954,98750,102975,201725
10 110,10,Shulin,42,1039,67810,90837,91474,182311
11 110,10,Yingge,20,434,32582,44114,44028,88142
12 110,10,Sanxia,28,608,44798,58042,58123,116165
13 110,10,Tamsui,42,698,84774,88496,96488,184984
14 110,10,Xizhi,50,1141,92166,100174,105691,205865
15 110,10,Ruifang,34,426,16121,19439,19147,38586
16 110,10,Wugu,20,515,35582,45351,45065,90416

```

<csvRead/>

讀取CSV檔案 – Print First Row

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LINE_SIZE 1024
int main(){
    /*Ex 9-8: read a csv file*/
    printf("Ex 9-8: read a csv file\n");
    // set filename
    char filename[] = "pop.csv";
    // open the file
    FILE *csv = fopen(filename, "r");
    // test whether the file can be opened
    if (csv==NULL) { // csv == NULL can be expressed as !csv
        fprintf(stderr, "failed to open file for reading\n");
        return 1;
    }
}
```

1. 因為平常在執行程式都是用終端機，萬一有錯誤，你也不知道是哪裡錯誤!
2. 所以我們需要在這個地方放一個stderr，如果無法打開檔案時，csv這個pointer會回傳NULL。
3. 此時你應該就要terminate這個程式，以免會有其他問題的發生。

</csvRead>

<csvRead/>

讀取CSV檔案 – Print First Row

```
// read data line-by-line  
char line[MAX_LINE_SIZE];
```

在讀取每一列資料的時候，最多能讀的字元數。
我們這邊MAX_LINE_SIZE為1024的字元。

```
// set a pointer for processing data  
char *result = NULL;
```

設定一個字元指標做為讀取資料時的掃描器

```
// move the starting pointer  
fseek(csv, 3, SEEK_SET);
```

移動一個字元指標讀取資料的起始點位置

```
// use fgets to fetch line data in csv  
fgets(line, MAX_LINE_SIZE, csv);
```

掃描一行資料進來

```
// print the fetched line data  
printf("%s", line);  
printf("-----\n");
```

印出這一行資料

</csvRead>

<csvRead/>

讀取CSV檔案 – Print First Row

```
// separate line data into string array by delimiter COMMA  
result = strtok(line, ",");
```

將delimiter(comma)做為分隔符號，此時就可以將字串切開來，變成以逗點為分割點的好幾個子字串。

```
// print line data  
printf("%s\n", line);  
printf("-----\n");
```

```
// parsing data  
for (int i=0;i<9;i++){  
    printf("%d\t", atoi(result));  
    result = strtok(NULL, ",");  
}
```

讀取資料

我們利用atoi()函數可以將字串轉整數。這邊有一個欄位是行政區，是一個字串。將comma忽略掉，變成NULL。

```
// close the file  
fclose(csv);
```

關閉檔案

}

</csvRead>

<csvRead/>

讀取CSV檔案 – Print All

如果我們需要調整程式碼，讓程式也可以讀取行政區的資料進來時，就會像前面一樣，要設定一個字串的array，將資料接進去存放。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LINE_SIZE 1024
int main(){
    /*Ex 9-9: read a csv file*/
    printf("Ex 9-9: read a csv file\n");
    // set filename
    char filename[] = "pop.csv";
    // open the file
    FILE *csv = fopen(filename, "r");
    // test whether the file can be opened
    if (!csv) {
        fprintf(stderr, "failed to open file for reading\n");
        return 1;
    }
    // read data line-by-line and
    // set a pointer for processing data
    char line[MAX_LINE_SIZE];
    char *result = NULL;
```

</csvRead>

<csvRead/>

讀取CSV檔案 – Print All

利用while loop掃描
每一列的資料

```
Ex 9-9: read a csv file
110 10 0 126 2515 219474 269207 284577 553784
110 10 0 119 2564 155491 187529 194797 382326
110 10 0 93 2998 169892 196965 210399 407364
110 10 0 62 1249 91708 102563 113921 216484
110 10 0 84 1842 162638 205512 216457 421969
110 10 0 69 1482 130834 144272 156609 300881
110 10 0 47 1236 90448 116819 120795 237614
110 10 0 38 727 73954 98750 102975 201725
110 10 0 42 1039 67810 90837 91474 182311
110 10 0 20 434 32582 44114 44028 88142
110 10 0 28 608 44798 58042 58123 116165
110 10 0 42 698 84774 88496 96488 184984
110 10 0 50 1141 92166 100174 105691 205865
110 10 0 34 426 16121 19439 19147 38586
110 10 0 20 515 35582 45351 45065 90416
110 10 0 17 466 29014 38251 39352 77603
110 10 0 17 443 50513 60308 63826 124134
110 10 0 10 195 16014 19962 20118 40080
110 10 0 8 239 9866 11824 11854 23678
110 10 0 12 107 3392 4102 3405 7507
110 10 0 7 78 2566 3608 3038 6646
110 10 0 13 254 9638 11502 10905 22407
110 10 0 9 124 4223 5847 5378 11225
110 10 0 15 201 7224 10241 10620 20861
110 10 0 10 186 7550 10772 10773 21545
110 10 0 12 120 2223 2433 1911 4344
110 10 0 12 244 3684 4490 3773 8263
110 10 0 11 202 4333 5914 5674 11588
110 10 0 5 43 1916 3118 3254 6372
[Finished in 230ms]
```

```
// move the starting pointer
```

```
fseek(csv, 3, SEEK_SET);
```

```
// use fgets to fetch line data in csv
```

```
while (fgets(line, MAX_LINE_SIZE, csv) != NULL){
```

```
// separate line data into string
```

```
// array by delimiter COMMA
```

```
result = strtok(line, ",");
```

```
// parsing data
```

```
int i;
```

```
for (i=0;i<9;i++){
```

```
    printf("%7d\t", atoi(result));
```

```
    result = strtok(NULL, ",");
```

```
}
```

```
printf("\n");
```

```
}
```

```
// close the file
```

```
fclose(csv);
```

```
}
```

</csvRead>

<csvRead/>

Read CSV Data into Variables

如果我們需要調整程式碼，讓程式也可以讀取行政區的資料進來時，就需要宣告一個字串的 **array**，將資料接進去存放。其他整數的資料，則可以使用 **int array** 儲存。

	Date	District	Data
1	110,10	Banqiao	126, 2515, 219474, 269207, 284577, 553784
2	110,10	Sanchong	119, 2564, 155491, 187529, 194797, 382326

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LINE_SIZE 1024
int main(){
    /*Ex 9-10: read a csv file*/
    printf("Ex 9-10: read a csv file\n");
    // set filename and open the file
    char filename[] = "pop.csv";
    FILE *csv = fopen(filename, "r");
    // test whether the file can be opened
    if (!csv) {
        fprintf(stderr, "failed to open file for reading\n");
        return 1;
    }
    // read data line-by-line
    // set a pointer for processing data
    // move the starting pointer
    char line[MAX_LINE_SIZE];
    char *result = NULL;
    fseek(csv, 3, SEEK_SET);
    int date[29][2] = {0};
    char disa[29][20];
    int popa[29][6] = {0};
    int i, j, row=0;
```

</csvRead>

Read CSV Data into Variables

```

// use fgets to fetch line data in csv
while (fgets(line, MAX_LINE_SIZE, csv) != NULL){
    // separate data with delimiter
    result = strtok(line, ",");
    // parsing data
    for (i=0; i<9; i++){
        if (i<2){
            date[row][i] = atoi(result);
        }else if(i==2){
            strcpy(disa[row], result);
        }else{
            popa[row][i-3] = atoi(result);
        }
        result = strtok(NULL, ",");
    }
    row++;
}
// close the file
fclose(csv);

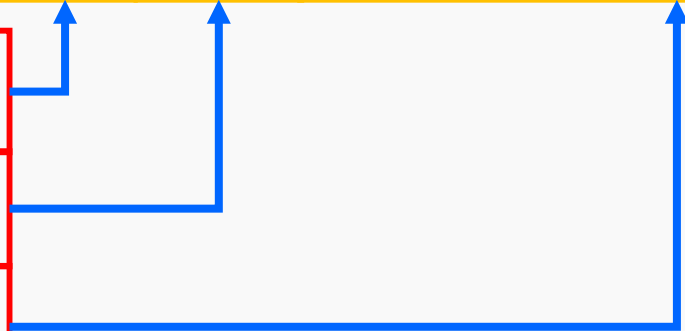
```

	Date	District	Data
1	110,10	Banqiao	126,2515,219474,269207,284577,553784
2	110,10	Sanchong	119,2564,155491,187529,194797,382326

```

if (i<2){
    date[row][i] = atoi(result);
}else if(i==2){
    strcpy(disa[row], result);
}else{
    popa[row][i-3] = atoi(result);
}

```



Read CSV Data into Variables

```

// print it out for testing
for (i=0;i<29;i++){
    for (j=0;j<9;j++){
        if (j<2){
            printf("%d\t", date[i][j]);
        }else if (j==2){
            printf("%10s\t", disa[i]);
        }else{
            printf("%7d\t", popa[i][j-3]);
        }
    }
}
printf("\n");
}
}

```

Result:

```

Ex 9-10: read a csv file
110 10    Banqiao    126    2515    219474    269207    284577    553784
110 10    Sanchong   119    2564    155491    187529    194797    382326
110 10    Zhonghe    93     2998    169892    196965    210399    407364
110 10    Yonghe     62     1249    91708     102563    113921    216484
110 10    Xinzhuang  84     1842    162638    205512    216457    421969
110 10    Xindian    69     1482    130834    144272    156609    300881
110 10    Tucheng    47     1236    90448     116819    120795    237614
110 10    Luzhou     38     727     73954     98750     102975    201725
110 10    Shulin     42     1039    67810     90837     91474     182311
110 10    Yingge     20     434     32582     44114     44028     88142
110 10    Sanxia     28     608     44798     58042     58123     116165
110 10    Tamsui     42     698     84774     88496     96488     184984
110 10    Xizhi      50     1141    92166     100174    105691    205865
110 10    Ruifang    34     426     16121     19439     19147     38586
110 10    Wugu       20     515     35582     45351     45065     90416
110 10    Taishan    17     466     29014     38251     39352     77603
110 10    Linkou     17     443     50513     60308     63826     124134
110 10    Bali       10     195     16014     19962     20118     40080
110 10    Shengkeng  8      239     9866      11824     11854     23678
110 10    Shiding    12     107     3392      4102      3405      7507
110 10    Pinglin    7      78      2566      3608      3038      6646
110 10    Sanzhi     13     254     9638      11502     10905     22407
110 10    Shimen     9      124     4223      5847      5378      11225
110 10    Jinshan    15     201     7224      10241     10620     20861
110 10    Wanli      10     186     7550      10772     10773     21545
110 10    Pingxi     12     120     2223      2433      1911      4344
110 10    Shuangxi   12     244     3684      4490      3773      8263
110 10    Gongliao   11     202     4333      5914      5674      11588
110 10    Wulai      5      43      1916      3118      3254      6372

```

<csvRead/>

Read CSV with Header

一般我們下載的資料都會有欄位說明的row，我們稱為header。這一次我們要讀取popWithHeader.csv

，除了將資料抓出來以外，我們也要將資料欄位的資訊一起擷取下來。

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LINE_SIZE 1024
int main(){
    /*Ex 9-11: read a csv file*/
    printf("Ex 9-11: read a csv file\n");
    // set filename and open the file
    char filename[] = "popWithHeader.csv";
    FILE *csv = fopen(filename, "r");
    // test whether the file can be opened
    if (!csv) {
        fprintf(stderr, "failed to open file for
        reading\n");
        return 1;
    }
    // read data line-by-line
    // set a pointer for processing data
    // move the starting pointer
    char line[MAX_LINE_SIZE];
    char *result = NULL;
    fseek(csv, 0, SEEK_SET);
    char field[9][20];
    int date[29][2] = {0};
    char disa[29][20];
    int popa[29][6] = {0};
    int i, j, row=0;
```

</csvRead>

Read CSV with Header

```
// use fgets to fetch line data in csv
while (fgets(line, MAX_LINE_SIZE, csv) != NULL){
    // separate data with delimiter
    result = strtok(line, ",");
    // parsing data
    for (i=0; i<9; i++){
        if (row==0){
            strncpy(field[i], result, strlen(result)+1);
        }else{
            if (i<2){
                date[row-1][i] = atoi(result);
            }else if(i==2){
                strcpy(disa[row-1], result);
            }else{
                popa[row-1][i-3] = atoi(result);
            }
        }
        result = strtok(NULL, ",");
    }
    row++;
}
fclose(csv); // close the file
```

<csvRead/>

Read CSV with Header

```
// print it out for testing
for (i=0;i<9;i++){
    if (i!=8){
        printf("%s\t", field[i]);
    }else{
        printf("%s", field[i]);
    }
}
for (i=0;i<29;i++){
    for (j=0;j<9;j++){
        if (j<2){
            printf("%7d\t", date[i][j]);
        }else if (j==2){
            printf("%10s\t", disa[i]);
        }else{
            printf("%7d\t", popa[i][j-3]);
        }
    }
    printf("\n");
}
}
```

Result:

```
Ex 9-11: read a csv file
year  month  district  village  neighborhood  house  men  women  total population
110   10     Banqiao  126     2515  219474  269207  284577  553784
110   10     Sanchong 119     2564  155491  187529  194797  382326
110   10     Zhonghe  93      2998  169892  196965  210399  407364
110   10     Yonghe   62      1249  91708  102563  113921  216484
110   10     Xinzhuang 84     1842  162638  205512  216457  421969
110   10     Xindian  69     1482  130834  144272  156609  300881
110   10     Tucheng  47     1236  90448  116819  120795  237614
110   10     Luzhou   38      727  73954  98750  102975  201725
110   10     Shulin  42     1039  67810  90837  91474  182311
110   10     Yingge   20      434  32582  44114  44028  88142
110   10     Sanxia   28      608  44798  58042  58123  116165
110   10     Tamsui   42      698  84774  88496  96488  184984
110   10     Xizhi    50     1141  92166  100174  105691  205865
110   10     Ruifang  34      426  16121  19439  19147  38586
110   10     Wugu     20      515  35582  45351  45065  90416
110   10     Taishan  17      466  29014  38251  39352  77603
110   10     Linkou   17      443  50513  60308  63826  124134
110   10     Bali     10      195  16014  19962  20118  40080
110   10     Shengkeng 8      239  9866  11824  11854  23678
110   10     Shiding  12      107  3392  4102  3405  7507
110   10     Pinglin  7       78  2566  3608  3038  6646
110   10     Sanzhi   13      254  9638  11502  10905  22407
110   10     Shimen   9       124  4223  5847  5378  11225
110   10     Jinshan  15      201  7224  10241  10620  20861
110   10     Wanli    10      186  7550  10772  10773  21545
110   10     Pingxi   12      120  2223  2433  1911  4344
110   10     Shuangxi 12      244  3684  4490  3773  8263
110   10     Gongliao 11      202  4333  5914  5674  11588
110   10     Wulai    5       43  1916  3118  3254  6372
```

</csvRead>

<Assignment/>

作業一

利用臺北市各行政區最新月份人口數及戶數(民國110年10月)，計算各行政區的男女佔總人口比例。

- (1) 首先先讀取CSV檔
- (2) 印出CSV檔內容

資料連結

<https://data.taipei/#/dataset/detail?id=6a1dbb4e-e99c-4e67-ab09-f6d83852dc99>

```
HW 9-1: Taipei City Population
```

```
---
month  district  village_door  village_house  neighborhood_door  neighborhood_house  house  population  men  women
10     Songshan   33            33             765                762                80296  194600      90991  103609
10     Xinyi       41            41             910                905                88127  208665      99058  109607
10     Daan        53            53            1026               1022              119724  292875     136535  156340
10     Zhongshan  42            42             871                869              100040  217277     100682  116595
10     Zhongzheng 31            31             583                582              65298  151326     71993  79333
10     Datong     25            25             523                521              52270  121468     58803  62665
10     Wanhua     36            36             722                722              78832  177799     86909  90890
10     Wenshan    43            43            1005               1002             107518  262791     125551  137240
10     Nangang    20            20             456                456              47972  116419     56717  59702
10     Neihu      39            39             916                916             110216  277880     132391  145489
10     Shilin     51            51             996                995             107618  271854     130977  140877
10     Beitou     42            42             823                821             98009  245345     117778  127567
```

<Assignment/>

作業一

- (3) 計算男女分別的人口占比
- (4) 再印出男女分別的人口占比

最後結果輸出應該為 →

District	Men Proportion	Women Proportion
Songshan	46.757965	53.242035
Xinyi	47.472264	52.527736
Daan	46.618865	53.381135
Zhongshan	46.338085	53.661915
Zhongzheng	47.574772	52.425228
Datong	48.410281	51.589719
Wanhua	48.880477	51.119523
Wenshan	47.775989	52.224011
Nangang	48.717993	51.282007
Neihu	47.643227	52.356773
Shilin	48.179170	51.820830
Beitou	48.005054	51.994946

參考資料

這邊有許多的延伸閱讀，有興趣的同學可以自己上去看看：

- (1) 檔案I/O的其他可能需要的防呆機制 [LINK](#)
- (2) 字串資料處理的內建函數庫 [LINK](#)