13

Case Study of File Processing

Case 1: Address Book

- Problem definition
  - An useful address book
    - Name
    - Phone Number
    - E-mail address
  - It includes two functions
    - Print address book (File processing)
    - Edit address book (Linked lists)

Source Code (1/4)

```
#include <iostream>
#include <fstream>

using namespace std;

void newdata(); // Key in new address data
void output(); // Output the address data to a file

struct node{
    char name[20];
    char phone_number[20];
    char e_mail[20];
    node *nxt;
};
node *start_ptr=NULL;
```

Source Code (2/4)

```
int main(){

    int select;

    do{
        select = 0; //Initialization
        cout << "Please key in..." << endl
             << "1. New address data" << endl
             << "2. Output the address data to a file" << endl
             << "3. Exit the program" << endl;
        cin >> select;

        if(select == 1) //key in new address data
            newdata();
        if(select ==2) //output the address data to a file
            output();
        cout << endl;

    }while(select != 3);
    return 0;
}
```

Source Code (3/4)

```
void output(){

    ofstream file("Address Book.txt", ios::app); // Output Address Book
    node *temp;

    temp = start_ptr;
    do{
        if(temp == NULL)
            file << "End of the list" << endl;
        else{
            file << "Name: " << temp->name << endl;
            file << "Phone Number: " << temp->phone_number << endl;
            file << "e-mail: " << temp->e_mail << endl << endl;
            temp=temp->nxt;
        }
    }while(temp!=NULL);
}
```

Source Code (4/4)

```
void newdata(){

    node *temp, *temp2;

    temp = new node;
    temp2=start_ptr;
    cout << "Please enter the name of the person: "; //key in name
    cin >> temp-> name;
    cout << "Please enter the phone number of the person: "; //key in phone number
    cin >> temp-> phone_number;
    cout << "Please enter the e-mail of the person: "; // key in e-mail
    cin >> temp->e_mail;
    temp -> nxt = NULL;

    if(start_ptr==NULL)
        start_ptr=temp;
    else{
        temp2=start_ptr;
        while(temp2->nxt != NULL)
            temp2=temp2->nxt;
        temp2->nxt=temp;
    }
}
```

Case 2: Print Map

Problem Definition

由提供的檔案 (**map.dat**)讀取資料，並且依規定格式顯示於螢幕。

0代表**wall** (以 **–** 及 **|** 表示)

1代表可行走之路徑 (以 ***** 表示)

2代表起始點 (以 **c** 表示)

0200
0110
0010
0010

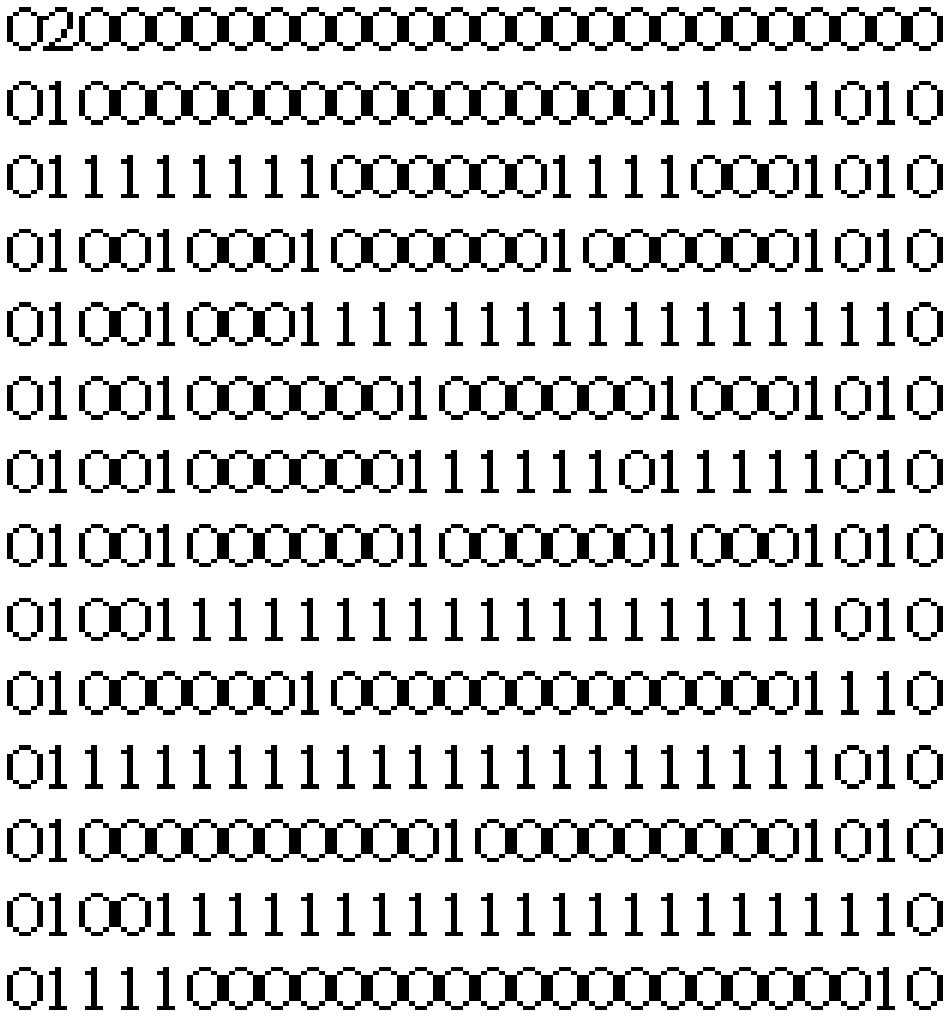
Source Code (1/2)

```
int main(int argc, char *argv[])
{
 FILE *file_open;
 char temp[100];
 int i;
 file_open = fopen("map.dat","r");
 if(file_open==NULL){
    printf("Can't open the file named map.dat"); //if error show the message
    fclose (file_open);
    }
 else{
      while(!feof(file_open)){
           fgets(temp,100,file_open);                    //一次從file 中讀取一行
           for(i=0;i!=strlen(temp);i++){
                if(temp[i]=='1')       printf("*"   //若 char=='1' 取代爲 *
                else if(temp[i]=='2') printf("c"   //若 char=='2' 取代爲 c
                else if(temp[i]=='0'){               //若 char=='0' 取代爲 wall
```

Source Code (2/2)

```
                    if((temp[i-1]=='0' && temp[i+1]=='0'))
                      printf("-");
                    else
                      printf("|");
                    }
                  }
              printf("\n");
              }
        }
 fclose(file_open);
    system("PAUSE");
    return EXIT_SUCCESS;
}
```

Data file



Result on the screen