

# 作業3

---

turnin code **cpp.hw3**

due date: **April 16, 23:59PM**

## 第1題

考慮**實例演練1**的程式，修改成為使用類別繼承以完成程式設計。參考下面的程式，我們定義了三種會員型態(以MemberType列舉型別宣告)，並簡化之前的Member類別設計：

```
#ifndef _MEMBER_
#define _MEMBER_

#include "name.h"

enum MemberType {Gold, Silver, Normal};

class Member
{
private:
    Name *name;
    string phone;
    MemberType type;

public:
    void showInfo();
    void setPhone(string str);
    Name *getName();
    void setName(Name *n);
    string getPhone();

    void setMemberType(MemberType mt);
    MemberType getMemberType();
};

class GoldMember : public Member
{
private:
    float discount;

public:
```

```
GoldMember(Member *m);
void setDiscount(float d);
float getDiscount();
void showInfo();
};

class SilverMember : public Member
{
private:
    int freeMinute;
public:
    SilverMember(Member *m);
    void setFreeMinute(int m);
    int getFreeMinute();
    void showInfo();
};
#endif
```

我們將上述的類別視為一般會員「Normal」並繼承「Member」類別完成「GoldMember」與「SilverMember」類別的設計。我們也先修改了「DataGetter」類別，使其能支援這三種類別的資料讀取：

```
Member * DataGetter::getMemberData(istream &is)
{
    Member *newMember=new Member;
    string str;

    getline(is, str);
    newMember->setName(new Name(str));
    getline(is, str);
    newMember->setPhone(str);
    getline(is, str);

    if(str.compare("Gold")==0)
    {
        newMember->setMemberType(Gold);
        GoldMember *newGoldMember = new GoldMember(newMember);
        getline(is, str);
        float discount;
        stringstream convert;
        convert << str;
        if(!(convert >> discount))
            discount=0.0;
        newGoldMember->setDiscount(discount);
        delete newMember;
        return newGoldMember;
    }
    else if(str.compare("Silver")==0)
    {
        newMember->setMemberType(Silver);
```

```
SilverMember *newSilverMember = new SilverMember(newMember);
getline(is, str);
int freeMinute;
stringstream convert;
convert << str;
if(!(convert >> freeMinute))
    freeMinute=0;
newSilverMember->setFreeMinute(freeMinute);
delete newMember;
return newSilverMember;
}
else if(str.compare("Normal")==0)
{
    newMember->setMemberType(Normal);
    return newMember;
}

return newMember;
}
```

```
#include <iostream>
#include <fstream>
#include "member.h"
#include "dataGetter.h"

using namespace std;

typedef Member* Members;

int main()
{
    Members *members = new Members[5];
    DataGetter *dg = new DataGetter;
    ifstream fin("members.dat");

    for(int i=0;i<5;i++)
    {
        members[i] = dg->getMemberData(fin);
    }

    fin.close();
    cout << "Input: ";
    string findName = dg->getAString(cin);
    bool found=false;
    while(findName.compare("Quit")!=0)
    {
```

```
for(int i=0;i<5;i++)
{
    if(((members[i]->getName())->get_lastname()).compare(findName)==0)
    {
        if(members[i]->getMemberType() == Gold)
        {
            ((GoldMember *)members[i])->showInfo();
        }
        else if(members[i]->getMemberType() == Silver)
        {
            ((SilverMember *)members[i])->showInfo();
        }
        else if(members[i]->getMemberType() == Normal)
        {
            members[i]->showInfo();
        }
        found=true;
    }
}
if(!found)
    cout << "Member not found!" << endl;
found=false;

cout << "Input: ";
findName = dg->getAString(cin);
}
cout << "Bye" << endl;
delete [] members;
}
```

相關檔案與Makefile皆已放置於`/home/stu/public/cpp/2015SCPPHW3/`目錄中，請複製到你的家目錄內以進行本作業。在該目錄中，已存在一個可執行檔`hw3demo`(注意，執行`hw3demo`時，必須在同一目錄內存在有`members.dat`測試檔)，你必須自行測試此程式，以瞭解本題要求的輸入及操作為何？並試著完成與`hw3demo`一樣的程式。

你必須完成一個名為`member2.cpp`的程式，其中包含了`GoldMember`與`SilverMember`類別的成員函式之實作，測試無誤後，請turning該檔案。 **本題僅需turnin `member2.cpp`檔案**

From:  
<https://junwu.nptu.edu.tw/dokuwiki/> - Jun Wu的教學網頁  
國立屏東大學資訊工程學系  
CSIE, NPTU  
Total: 173440

Permanent link:  
<https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=cpp:homework:hw3>

Last update: **2019/07/02 15:01**

