



中山大学
SUN YAT-SEN UNIVERSITY

实验报告

实验人：黄敏怡

学号：15331115

日期：2015.5.27

院（系）：数据科学与计算机学院

专业（班级）：软件学院3班

实验题目：校园电子卡管理系统

一.实验目的

本实验面向C++语言的初学者。

主要让实验者熟悉面向对象的编程思想以及类的使用。

二.实验环境

本实验可基于 Xcode 平台开发，参考主流的编码规范，如[Google C++Style Guide](#)（[中文版](#)）

2.1 编程语言和开发工具

编程语言：C++

开发工具：Xcode

2.2 编码规范

要求遵循良好的程序设计风格来设计和编写程序。基本编码规范：

1. 标识符的命名要到达顾名思义的程度；
2. 关键代码提供清晰、准确的注释；
3. 程序版面要求：
 - a) 不同功能块用空行分隔；
 - b) 一般一个语句一行；
 - c) 语句缩进整齐、层次分明。

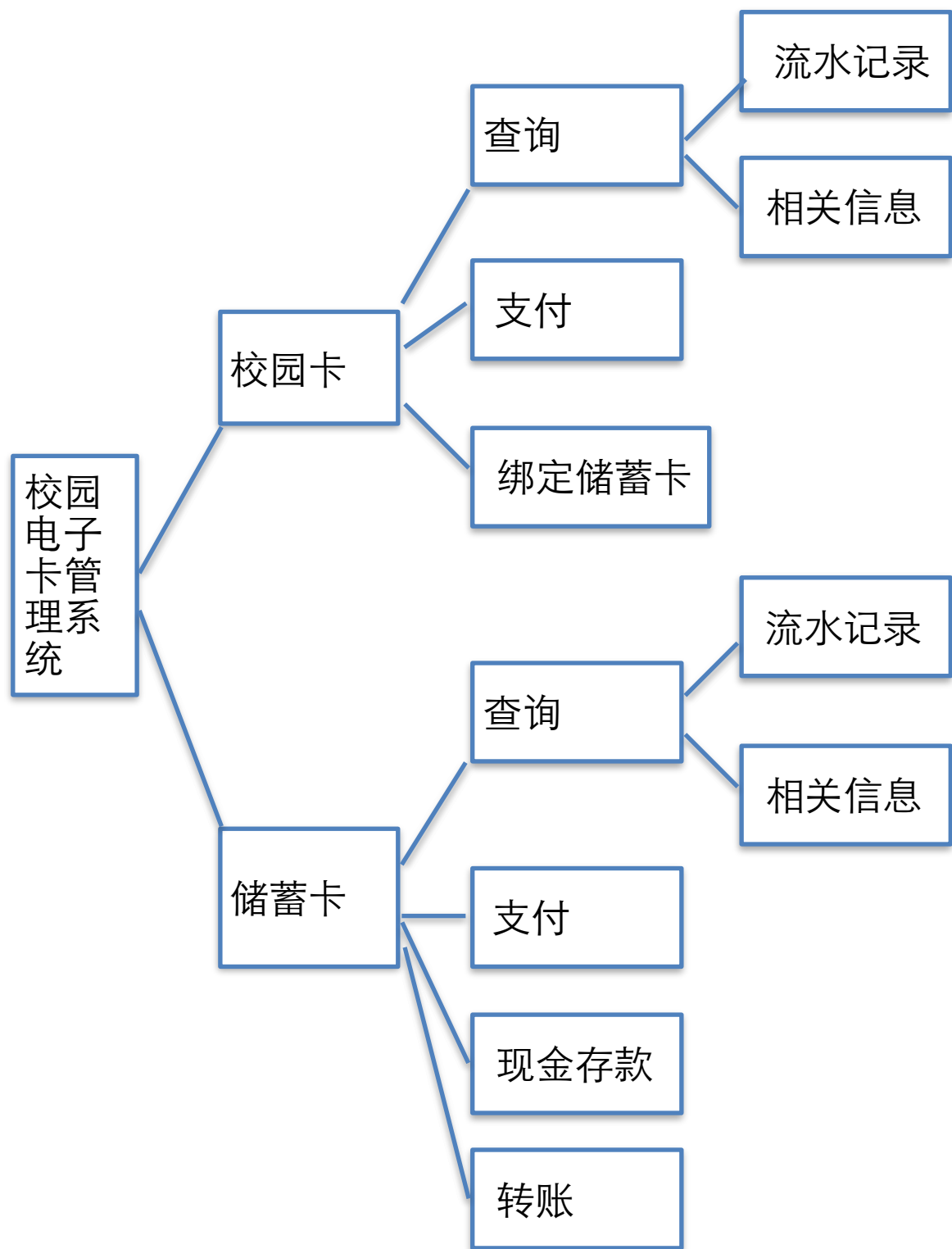
三.实验内容

通过c++语言实现对校园电子卡系统的管理，设计程序实现校园卡与储蓄卡支付消费查询转账等日常功能，并能使用户拥有良好体验（余额不足提醒，透支提醒，界面友好）。

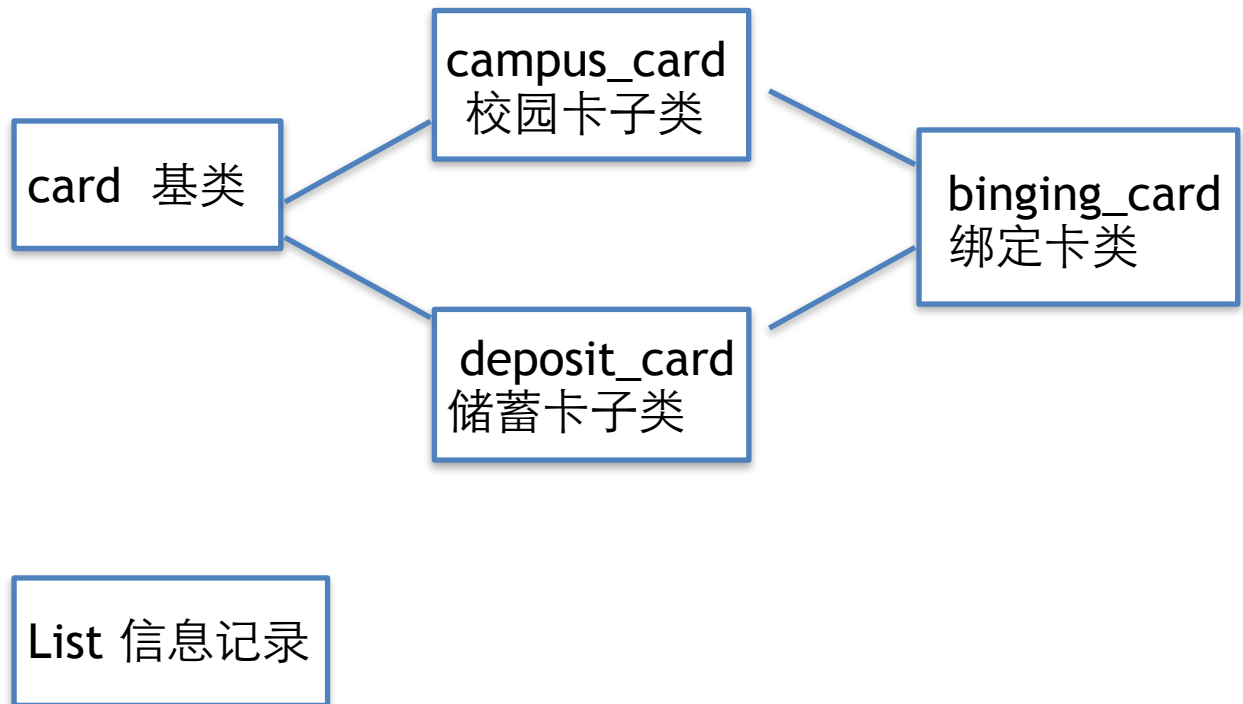
四.分析与设计

简要描述程序设计的过程，包括设计思路，设计要点及特色；程序的不足与改进等。要求画出程序的简单流程图。

4.1、需求分析：



4.2、类结构设计



4.3、细节设计

接口设计：

```
int getYear() {return year; }
int getMon() {return mon; }
int getDay() {return day; }
```

数据成员设计：

```
class List { //流水信息记录
```

```
public:
```

```
    int flag1 = 0; //记录 1： 何时何地消费多少
```

```

int flag2 = 0;//记录 2 : 何时转账给某账户多少钱
int flag3 = 0;//记录 3 : 何时某账户转账给自己多少钱
int flag4 = 0;//记录 4 : 何时现金存款多少钱
string adress;//消费地点
double amt;//消费金额
long _id;//转账账户
//消费时间
int _year;
int _mon;
int _day;
int hour;
int min;
int sec;
};

```

```

class card { //基类
public:
    card() {}
    virtual void pay() { //支付函数
    }
    virtual void query() { //查询函数
    }
    long id;//卡号
    string name;//持卡人姓名
    double balance;//余额
    int list_num;//流水信息数组的个数
    List *ListArray;//流水信息数组
};

```

```

class campus_card : public card {

```

```

public:
    campus_card();//创建一个校园卡账户
    void pay(string _adress, double amount);
    void query();//查询 1 校园卡相关信息 2 校园卡的流水记录信息
    string college;//学院
    bool isBind;//是否绑定储蓄卡
    long bindID;//绑定的储蓄卡的卡号
};

class deposit_card : public card {
public:
    deposit_card();//创建一个储蓄卡账户
    void pay(string _adress, double amount); //用该储蓄卡支付，若支付的金额
    超过余额，多出的部分则会自动选择透支，若超过透支余额则操作失败
    void query();//查询 1 储蓄卡相关信息 2 支付收款流水记录信息
    void transfer1(deposit_card &othercard);//转账给其他储蓄卡
    void transfer2(campus_card &othercard);//转账给其他校园卡
    void deposit(double amount);//现金存款
    int getYear() {return year; }
    int getMon() {return mon; }
    int getDay() {return day; }
private:
    //发卡日期
    int year;
    int mon;
    int day;
    double overdraft_limit;//总的透支额度，创建账户时会将透支额度设为1000
    元
    double overdraft_money;//已被透支的额度
    double overdraft_rest;//剩余的可透支额度

```

```
};
```

```
class binding_card : public campus_card, public deposit_card {  
public:
```

```
    void pay(string _adress, double amount, deposit_card *card1, int id1);//当校园  
卡余额不足时，多出的部分则直接使用储蓄卡支付  
};
```

成员函数：

```
virtual void pay() { //支付函数
```

```
}
```

```
virtual void query() { //查询函数
```

```
}
```

```
campus_card(); //创建一个校园卡账户
```

```
void pay(string _adress, double amount);
```

```
void query();
```

```
deposit_card(); //创建一个储蓄卡账户
```

```
void pay(string _adress, double amount); //用该储蓄卡支付，若支付的金额  
超过余额，多出的部分则会自动选择透支，若超过透支余额则操作失败
```

```
void query(); //查询 1 储蓄卡相关信息 2 支付收款流水记录信息
```

```
void transfer1(deposit_card &othercard); //转账给其他储蓄卡
```

```
void transfer2(campus_card &othercard); //转账给其他校园卡
```

```
void deposit(double amount); //现金存款
```

```
void pay(string _adress, double amount, deposit_card *card1, int id1); //当  
校园卡余额不足时，多出的部分则直接使用储蓄卡支付
```

数据成员：

```
int flag1 = 0; //记录 1：何时何地消费多少
```

```
int flag2 = 0;//记录 2 : 何时转账给某账户多少钱
int flag3 = 0;//记录 3 : 何时某账户转账给自己多少钱
int flag4 = 0;//记录 4 : 何时现金存款多少钱
string adress;//消费地点
double amt;//消费金额
long _id;//转账账户
//消费时间
int _year;
int _mon;
int _day;
int hour;
int min;
int sec;
int year;
int mon;
int day;
double overdraft_limit;//总的透支额度，创建账户时会将透支额度设为
1000元
double overdraft_money;//已被透支的额度
double overdraft_rest;//剩余的可透支额度
```

五.实验结果

设计有代表性的输入数据，并分析测试结果以及针对测试中发现的问题所进行的

1.创建两个新储蓄卡账户

主界面	
1	创建一个新的储蓄卡账户
2	创建一个新的校园卡账户
3	储蓄卡账户登录
4	校园卡账户登录
5	显示所有储蓄卡
6	显示所有校园卡
7	退出

请输入您的选择：1

请输入户主的姓名：hmy1

请输入该账户的卡号：10001000

请输入您为该账户充值的金额：100

储蓄卡账户创建成功！系统已自动为该账户设置**1000**元的可透支金额。

主界面	
1	创建一个新的储蓄卡账户
2	创建一个新的校园卡账户
3	储蓄卡账户登录
4	校园卡账户登录
5	显示所有储蓄卡
6	显示所有校园卡
7	退出

请输入您的选择：1

请输入户主的姓名：hmy2

请输入该账户的卡号：20002000

请输入您为该账户充值的金额：0

储蓄卡账户创建成功！系统已自动为该账户设置**1000**元的可透支金额。

2.创建新校园卡账户

主界面	
1	创建一个新的储蓄卡账户
2	创建一个新的校园卡账户
3	储蓄卡账户登录
4	校园卡账户登录
5	显示所有储蓄卡
6	显示所有校园卡
7	退出

请输入您的选择：2

请输入您所在的学院名称：pc

请输入您的姓名：hmy

请输入您的学号：15331116

请输入您为该校园卡充值的金额：0

校园卡账户创建成功！

3.储蓄卡账户登录

(1) 存款与支付

储蓄卡功能界面	
1	现金存款
2	支付功能
3	转账功能
4	查询功能
5	退出

请输入您的选择：1
请输入您所存入的金额：100
存款成功！余额为200元

储蓄卡功能界面	
1	现金存款
2	支付功能
3	转账功能
4	查询功能
5	退出

请输入您的选择：2
请输入消费地点：carteen
请输入支付金额：50
支付成功！
您的余额为150元

储蓄卡功能界面	
1	现金存款
2	支付功能
3	转账功能
4	查询功能
5	退出

(2) 转账

请输入您的选择：3

1	转账给其他储蓄卡
2	转账给其他校园卡

请输入您的选择：1
请输入所要转入的储蓄卡的卡号：20002000
请输入转账的金额：50
转账成功！
储蓄卡余额为100元
转账成功！

储蓄卡功能界面	
1	现金存款
2	支付功能
3	转账功能
4	查询功能
5	退出

请输入您的选择：3

1	转账给其他储蓄卡
2	转账给其他校园卡

请输入您的选择：2
请输入所要转入的校园卡的卡号：15331116
请输入转账的金额：50
转账成功！
储蓄卡余额为50元

(3) 查询

储蓄卡功能界面	
1	现金存款
2	支付功能
3	转账功能
4	查询功能
5	退出

请输入您的选择：4

1	储蓄卡相关信息
2	支付收款流水记录信息

请输入您的选择：1

卡号	发卡日期	持卡人姓名	余额
10001000	2016-5-26	hmy1	50
可透支额度	已透支的金额	剩余可透支额度	
1000	0	1000	

1	储蓄卡相关信息
2	支付收款流水记录信息

请输入您的选择：2

卡号为10001000的hmy1用户，您的储蓄卡流水记录信息如下		
2016-5-26 16:10:24	现金存款	转入100元
2016-5-26 16:10:24	您在carteen	消费了50元
2016-5-26 16:10:24	您给卡号为20002000的账户	转账了50元
2016-5-26 16:10:24	您给卡号为15331116的账户	转账了50元

4.校园卡账户登录

(1) 绑定

主界面	
1	创建一个新的储蓄卡账户
2	创建一个新的校园卡账户
3	储蓄卡账户登录
4	校园卡账户登录
5	显示所有储蓄卡
6	显示所有校园卡
7	退出

请输入您的选择：4
请输入您的学号：15331116
登录成功！请选择您要对该校园卡进行的操作。

校园卡功能界面	
1	支付功能
2	查询功能
3	绑定储蓄卡
4	退出

请输入您的选择：3
请输入所要绑定的储蓄卡的卡号：10001000
绑定成功！

(2) 支付

校园卡功能界面	
1	支付功能
2	查询功能
3	绑定储蓄卡
4	退出

请输入您的选择：1
请输入消费地点：book
请输入支付金额：100
校园卡余额不足，多出部分由绑定卡支付
支付成功！
您的余额为0元

校园卡功能界面	
1	支付功能
2	查询功能
3	绑定储蓄卡
4	退出

请输入您的选择：1
请输入消费地点：room
请输入支付金额：200
校园卡余额不足，多出部分由绑定卡支付
储蓄卡余额不足，使用透支额度支付成功！
您还可以透支850元

(3) 查询及绑定多张卡

校园卡功能界面	
1	支付功能
2	查询功能
3	绑定储蓄卡
4	退出

请输入您的选择：3
请输入所要绑定的储蓄卡的卡号：20002000
绑定成功！

校园卡功能界面	
1	支付功能
2	查询功能
3	绑定储蓄卡
4	退出

请输入您的选择：2

1	校园卡相关信息
2	校园卡流水记录

请输入您的选择：1

学号	学院	姓名	余额
15331116	pc	hmy	0

是否绑定储蓄卡：是。

学号为15331116的hmy同学，您的校园卡流水记录信息如下

2016-5-26 16:10:24	卡号为10001000的储蓄卡账户给您转账了50元
2016-5-26 16:10:24	您在book消费了50元

5.显示所有储蓄卡

请输入您的选择：5

卡号	发卡日期	持卡人姓名	余额
10001000	2016-5-26	hmy1	0
20002000	2016-5-26	hmy2	50

6.显示所有校园卡

请输入您的选择：6

学号	学院	姓名	余额
15331116	pc	hmy	0

六.设计心得

进行总结，描述所获得的经验和心得体会等。

本次实验项目感觉与之前的项目有些许相似之处，但仍学到不少东西。

（1）本来日期是想让用户自己输入，但是体验不好，后采用<ctime>库的函数解决了这个问题。

（2）一开始想采用文件输入输出的方式记录，但由于个人能力问题，运行时出现了各种内存错误，最终放弃，决心好好学文件。

（3）校园卡系统与实际生活很相近，在实现基本功能后决定添加一些辅助小功能，例如每次消费后会显示余额，用了透支额度会告知，不然透支了都不知道。

（4）每个项目之间都有可借取的地方，可以看看自己以前的代码，缩进改良一下。

（5）界面友好还是很重要的，花了有点时间调整输出界面，相信以后会熟练很多。

注：提示信息

1. 可发挥想象力，提高系统的交互体验，可用性和可靠性。
2. 请独立完成，但允许和鼓励相互讨论后自己写程序。由于题目具有开放性，很容易从程序看出是否抄袭。若发现抄袭，将导致抄袭双方或多方大副扣分或直接零分。

程序测试

1. 请自行测试类的功能。
2. 界面交互时，请检测用户输入的正确性，保证程序的健壮性。（在用户做出操作后最好打印出提示语句。）
3. 在实验报告中分析测试意图和测试结果。如果通过测试发现程序错误，也分析错误原因、改正方法等，并在实验报告中体现。