面向对象程序设计 C++ 第 6 次实验

题目1

代码思路

创建一个Time类，内含三个int型变量hour,minute,second，分别代表当前时间的时分秒。

```
class Time
{
private:
    int hour, minute, second;
};
```

增加int toSecond()方法：将当前Time转化为一个int型的秒数返回。

```
int toSecond()const{
    return hour * 3600 + minute * 60 + second;
}
```

增加Time toTime(int s)方法：将一个特定的秒数，转化为Time类型，转化前先将s对86400求余转化为一个0-86400正整数，确保

```
Time toTime(int s)const {
    s = ((s % 86400) + 86400) % 86400;
    return Time(
        s / 3600,
        s % 3600 / 60,
        s % 60
    );
}
```

实现方法bool operator ==(const Time& t)const;

将三个变量分别对比即可。

```
bool operator ==(const Time& t)const {
    return (hour == t.hour) && (minute == t.minute) && (second == t.second);
}
```

实现方法operator !=(const Time& t)const;

调用==的操作符重载方法并将结果取反即可。

```
bool operator !=(const Time& t)const {
    return !(*this == t);
}
```

实现方法bool operator >(const Time& t)const;

调用toSecond()方法比较两者秒数的大小即可。

```
bool operator >(const Time& t)const {
    return toSecond() > t.toSecond();
}
```

实现方法bool operator >=(const Time& t)const;

调用>的重载和==的重载，两者取或即可。

```
bool operator >=(const Time& t)const {
    return *this > t || *this == t;
}
```

实现方法bool operator <(const Time& t)const;

<即>取反且!=。

```
bool operator <(const Time& t)const {
    return !(*this > t) && (*this != t);
}
```

实现方法bool operator <=(const Time& t)const;

将>取反即可。

```
bool operator <=(const Time& t)const {
    return !(*this > t);
}
```

实现方法Time operator +=(int s)const;

调用toSecond()转化为秒数加上新增加的秒数再调用toTime()转化为Time并返回。

```
Time operator +=(int s)const {
    return toTime(toSecond() + s);
}
```

实现方法Time operator -=(int s)const;

调用toSecond()转化为秒数减去新减少的秒数再调用toTime()转化为Time并返回。

```
Time operator -=(int s)const {
    return toTime(toSecond() - s);
}
```

实现方法Time operator --()const;和Time operator ++()const;

调用上面两个重载即可。

```
Time operator ++()const {
    return *this += 1;
}
Time operator --()const {
    return *this -= 1;
}
```

实现方法int operator -(const Time& t)const;

转化为秒数相减即可。

```
int operator -(const Time& t)const {
    return toSecond() - t.toSecond();
}
```

程序运行结果

主函数

```
int main() {
    Time t1 = Time(1, 2, 5);
    Time t2 = Time(2, 1, 4);
    if (t2 > t1)cout << ">" << endl;
    (t1 -= 5999).display();
    (t2 += 3600).display();
    (--t1).display();
    cout << t1 - t2 << endl;
}
```

运行结果

题目2

代码思路

在User的变量中加上ComputerLab的友元类

后可以将get_name()注释掉，因为在ComputerLab类中已经可以直接取得User的属性。

```
class User
{
public:
    User() {
        this->name = "empty";
    }

    User(string name) {
        this->name = name;
    }

    //string get_name() {
    //  return name;
    //}

private:
    string name;
    friend class ComputerLab;
};
```

改写后的login()

```
bool login(User user, int x, int y) {
    if (x > computers.size() || y > computers[x - 1].size() || is_user_exist(user.name) ||
computers[x - 1][y - 1].name != "empty") {
        cout << "invalid login" << endl;
        return 0;
    }
    computers[x - 1][y - 1] = user;
    return 1;
}
```

实现+的运算符重载，只需调用login()即可

```
void operator+(loginReq &r) {
    r.flag = login(*r.userPointer, r.labNum, r.stationNum);
}
```

改写后的logout()

```
bool logout(User user) {
    for (int i = 0; i < computers.size(); i++) {
        for (int j = 0; j < computers[i].size(); j++) {
            if (computers[i][j].name == user.name) {
                computers[i][j] = User();
                return 1;
            }
        }
    }
    cout << "invalid logoff" << endl;
    return 0;
}
```

实现-的重载只需调用logout()即可

```
void operator-(logoffReq& r) {
    r.flag = logout(*r.userPointer);
}
```

改写main函数

```
int main() {
    ComputerLab computerLab;
    while (1) {
        string s;
        cin >> s;
        if (s == "+") {
            string name;
            int x, y;
            cin >> name >> x >> y;
            loginReq r = { new User(name), x, y };
            //computerLab.login(name, x, y);
            computerLab + r;
        }
        else if (s == "-") {
            string name;
            cin >> name;
            logoffReq r = { new User(name) };
            //computerLab.logout(name);
            computerLab - r;
        }
        else if (s == "=") {
            break;
        }
        computerLab.print();
    }
}
```

程序运行结果

Microsoft Visual Studio 调试控

```
+ SWE100 1 1
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: empty
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: empty
+ DMT200 2 6
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: empty
+ SWE400 1 1
invalid login
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: empty
+ SWE400 4 3
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
+ SWE400 2 1
invalid login
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
+ SWE700 2 3
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: SWE700 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
- SWE700
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
- DMT700
invalid logoff
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
+ SWE800 1 6
invalid login
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
```

```
+ SWE900 5 1
invalid login
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
- SWE700
invalid logoff
1 1: SWE100 2: empty 3: empty 4: empty 5: empty
2 1: empty 2: empty 3: empty 4: empty 5: empty 6: DMT200
3 1: empty 2: empty 3: empty 4: empty
4 1: empty 2: empty 3: SWE400
=

C:\Users\CC507\source\repos\C语言\Project3\x64\Debug\Project3
按任意键关闭此窗口. . .
```

题目3

代码思路

新增一个Date类便于存储和计算日期

包含三个int型变量day,month,year分别表示日，月，年

包含无参构造函数和全参构造函数和一个使用字符串(形如1/1/2025)进行初始化的构造函数

toString()：将Date转化为字符串

culDays()：计算当前日期与参数日期的相差天数

```
class Date {
public:
    Date() {
        this->month = 0;
        this->day = 0;
        this->year = 0;
    }
    Date(int day, int month, int year) {
        this->month = month;
        this->day = day;
        this->year = year;
    }
    Date(string s) {
        istringstream iss(s);
        string token;
        getline(iss, token, '/');
        day = stoi(token);
        getline(iss, token, '/');
        month = stoi(token);
        getline(iss, token);
        year = stoi(token);
```

```
    }
    string toString() {
        string s;
        s += static_cast<char>(day / 10 + '0');
        s += static_cast<char>(day % 10 + '0');
        s += "/";
        s += static_cast<char>(month / 10 + '0');
        s += static_cast<char>(month % 10 + '0');
        s += "/";
        s += static_cast<char>(year / 1000 + '0');
        s += static_cast<char>(year / 100 % 10 + '0');
        s += static_cast<char>(year / 10 % 10 + '0');
        s += static_cast<char>(year % 10 + '0');
        return s;
    }
    int culDays(Date date) {
        int months[12] = { 31,28,31,30,31,30,31,31,30,31,30,31 };
        int count = 0;
        if ((year % 400 == 0) || ((year % 4 == 0) && (year % 100 != 0)))months[1] = 29;
        else months[1] = 28;
        count += months[month - 1] - day;
        for (int i = month; i < 12; i++) {
            count += months[i];
        }
        for (int i = year + 1; i < date.year; i++) {
            if ((i % 400 == 0) || ((i % 4 == 0) && (i % 100 != 0)))count += 366;
            else count += 365;
        }
        if ((date.year % 400 == 0) || ((date.year % 4 == 0) && (date.year % 100 != 0)))months[1]
= 29;
        else months[1] = 28;
        for (int i = 0; i < date.month - 1; i++) {
            count += months[i];
        }
        count += date.day;
        return count;
    }
private:
    int month, day, year;
};
```

为User类添加一个Date变量，虚函数culPrice()用于计算上机所需要的价格。

```
class User
{
public:
    User() {
        this->name = "empty";
        this->date = Date();
    }

    User(string name, Date date) {
        this->name = name;
        this->date = date;
    }

    virtual int culPrice(Date logoffdate) {
```

```
        return 0;
    }

protected:
    string name;
    Date date;
    friend class ComputerLab;
};
```

在其两个子类中分别重写culPrice()价格计算方法

```
class Student :public User {
public:

    Student(string name, Date date) {
        this->name = name;
        this->date = date;
    }

    int culPrice(Date logoffdate) override{
        int days = date.culDays(logoffdate);
        if (days <= 14)return 0;
        else return days - 14;
    }

};

class Staff :public User {
public:

    Staff(string name, Date date) {
        this->name = name;
        this->date = date;
    }

    int culPrice(Date logoffdate) override{
        int days = date.culDays(logoffdate);
        if (days <= 30)return 2 * days;
        else return 2 * 30 + 4 * (days - 30);
    }
};
```

四种请求传参的设计：

StaOffReq和StuOffReq中User子类的Date字段会传入登出时间

```
struct StaInReq {
    Staff* userPointer;
    int labNum;
    int stationNum;
    bool flag;
};

struct StuInReq {
    Student* userPointer;
    int labNum;
    int stationNum;
```

```
    bool flag;
};

struct StaOffReq {
    Staff* userPointer;
    bool flag;
};

struct StuOffReq {
    Student* userPointer;
    bool flag;
};
```

login()方法的设计以及+的操作符重载

```
bool login(User user, int x, int y) {
    if (x > computers.size() || y > computers[x - 1].size() || is_user_exist(user.name) ||
computers[x - 1][y - 1].name != "empty") {
        cout << "invalid login" << endl;
        return 0;
    }
    computers[x - 1][y - 1] = user;
    return 1;
}

void operator+(StuInReq& r) {
    r.flag = login(*r.userPointer, r.labNum, r.stationNum);
}

void operator+(StaInReq& r) {
    r.flag = login(*r.userPointer, r.labNum, r.stationNum);
}
```

logoff的设计以及-的操作符重载

登出时需要判断User的类型，调用对应的方法计算价格

```
bool logout(User user) {
    for (int i = 0; i < computers.size(); i++) {
        for (int j = 0; j < computers[i].size(); j++) {
            if (computers[i][j].name == user.name) {
                if (computers[i][j].name[0] >= '0' && computers[i][j].name[0] <= '9')
                    cout << computers[i][j].name << " log off, " <<
                    "time: " << computers[i][j].date.culDays(user.date) << "days, " <<
                    "price: " << Staff(computers[i][j].name, computers[i]
[j].date).culPrice(user.date) << endl;
                else
                    cout << computers[i][j].name << " log off, " <<
                    "time: " << computers[i][j].date.culDays(user.date) << " days, " <<
                    "price: " << Student(computers[i][j].name, computers[i]
[j].date).culPrice(user.date) << endl;
                computers[i][j] = User();
                return 1;
            }
        }
    }
    cout << "invalid logoff" << endl;
```

```
    return 0;
}

void operator-(StaOffReq& r) {
    r.flag = logout(*r.userPointer);
}

void operator-(StuOffReq& r) {
    r.flag = logout(*r.userPointer);
}
```

main函数的设计：

登入时需要判断name的类型并根据name创造不同类型的User录入ComputerLab

```
int main() {
    ComputerLab computerLab;
    while (1) {
        string s;
        cin >> s;
        if (s == "+") {
            string name, date;
            int x, y;
            cin >> name >> x >> y >> date;
            if (name[0] >= '0' && name[0] <= '9') {
                StaInReq r = { new Staff(name,Date(date)), x, y };
                computerLab + r;
            }
            else {
                StuInReq r = { new Student(name,Date(date)), x, y };
                computerLab + r;
            }
        }
        else if (s == "-") {
            string name, date;
            cin >> name >> date;
            StaOffReq r = { new Staff(name,Date(date)) };
            computerLab - r;
        }
        else if (s == "=") {
            break;
        }
        computerLab.print();
    }
}
```

程序运行结果

Microsoft Visual Studio 调试控

```
+ SWE100 1 1 1/1/2016
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: empty 4: empty 5: empty 6: empty
+ DMT200 2 6 02/04/2016
invalid login
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: empty 4: empty 5: empty 6: empty
+ SWE400 1 1 1/01/2016
invalid login
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: empty 4: empty 5: empty 6: empty
+ SWE400 4 3 10/1/2016
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
+ SWE400 2 1 1/1/2015
invalid login
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
+ 2019007 2 3 1/1/2015
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: 2019007 01/01/2015 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
- 2019007 1/12/2016
2019007 log off, time: 700days, price: 2740
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
- DMT700 1/12/2016
invalid logoff
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
+ SWE800 1 6 10/10/2013
invalid login
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
+ SWE900 5 1 10/10/2014
invalid login
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
- SWE700 1/12/2016
invalid logoff
1 1: SWE100 01/01/2016 2: empty 3: empty
2 1: empty 2: empty 3: empty 4: empty
```

```
3 1: empty 2: empty 3: empty 4: empty 5: empty
4 1: empty 2: empty 3: SWE400 10/01/2016 4: empty 5: empty 6: empty
=

C:\Users\CC507\source\repos\C语言\Project3\x64\Debug\Project3.exe (进程 5494
按任意键关闭此窗口. . .
```