单链表及学生管理

1. 单链表实现

代码略。详见源程序。

main.cpp

#include <stdio.h>

#include "LinkList.cpp"

int main**()**

**{**

int a**[**10**];**

**for** **(**int i**=**0**;** i**<**10**;** i**++)**

**{**

a**[**i**]** **=** i**;**

**}**

int b**[**10**];**

**for** **(**int i**=**0**;** i**<**10**;** i**++)**

**{**

a**[**i**]** **=** 2**\***i**;**

**}**

LinkList**<**int**>** L1**(**a**,**10**);**

cout**<<**"Print all list:"**<<**endl**;**

L1**.**PrintLinkList**();**

cout**<<**"Get length:"**<<**endl**;**

cout**<<**L1**.**ListLength**()<<**endl**;**

cout**<<**"Get position 3:"**<<**endl**;**

int s **=** L1**.**Get**(**3**);**

cout**<<**s**<<**endl**;**

cout**<<**"Find value 8:"**<<**endl**;**

cout**<<**L1**.**Locate**(**8**)<<**endl**;**

cout**<<**"Insert position 2 and delete position 5:"**<<**endl**;**

L1**.**Insert**(**2**,**13**);**

L1**.**Delete**(**5**);**

L1**.**PrintLinkList**();**

cout**<<**"Invert:"**<<**endl**;**

L1**.**Invert**();**

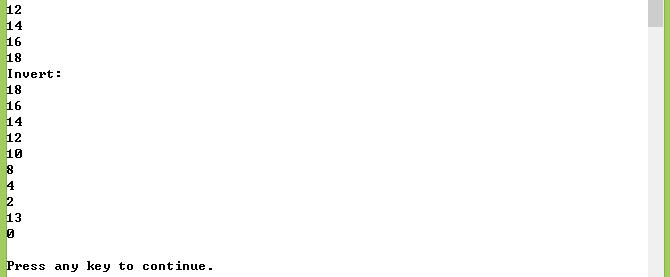
L1**.**PrintLinkList**();**

**return** 0**;**

**}**

测试截图：





单链表的学生管理

Student.cpp

#ifndef STUDENT\_H

#define STUDENT\_H

#include <string>

#include <iostream>

**using** **namespace** std**;**

class Student

**{**

private**:**

int number**;** //学号

string name**;** //姓名

char gender**;** //性别

string major**;** //专业

int birthyear**;** //出生年

int birthmonth**;** //出生月

public**:**

Student**();**

**~**Student**();**

friend ostream **&** **operator<<(**ostream **&** os**,** Student **&** s1**);** //重载输出流

friend istream **&** **operator>>(**istream **&** is**,** Student **&** s1**);** //重载输入流

friend bool **operator!=(**Student **&** s1**,**int num**);**

int getNumber**(**Student s**);**

string getName**(**Student s**);**

char getGender**(**Student s**);**

string getMajor**(**Student s**);**

int getBirthyear**(**Student s**);**

int getBirthmonth**(**Student s**);**

void setNumber**(**int number**);**

void setName**(**string name**);**

void setGender**(**char gender**);**

void setMajor**(**string major**);**

void setBirthyear**(**int birthyear**);**

void setBirthmonth**(**int birthmonth**);**

**};**

#endif // STUDENT\_H

Student.cpp

#include "Student.h"

#include <iostream>

**using** **namespace** std**;**

Student**::**Student**()**

**{**

**}**

Student**::~**Student**()**

**{**

**}**

ostream **&** **operator<<(**ostream **&** os**,**Student **&** s1**)**

**{**

os**<<**"Number:"**<<**s1**.**getNumber**(**s1**)<<**endl**;**

os**<<**"Name:"**<<**s1**.**getName**(**s1**)<<**endl**;**

os**<<**"Gender:"**<<**s1**.**getGender**(**s1**)<<**endl**;**

os**<<**"Major:"**<<**s1**.**getMajor**(**s1**)<<**endl**;**

os**<<**"Birthyear:"**<<**s1**.**getBirthyear**(**s1**)<<**endl**;**

os**<<**"Birthmonth:"**<<**s1**.**getBirthmonth**(**s1**)<<**endl**;**

**return** os**;**

**}**

istream **&** **operator>>(**istream **&** is**,**Student **&** s1**)**

**{**

cout **<<** "Number:"**;**

is **>>** s1**.**number**;**

cout **<<** "Name:"**;**

is **>>** s1**.**name**;**

cout **<<** "Gender:"**;**

is **>>** s1**.**gender**;**

cout **<<** "Major:"**;**

is **>>** s1**.**major**;**

cout **<<** "Birthyear:"**;**

is **>>** s1**.**birthyear**;**

cout **<<** "Birthmonth:"**;**

is **>>** s1**.**birthmonth**;**

**return** is**;**

**}**

bool **operator!=(**Student **&** s1**,**int num**)**

**{**

**return** **(**s1**.**number**!=**num**);**

**}**

int Student**::**getNumber**(**Student s**)**

**{**

**return** number**;**

**}**

string Student**::**getName**(**Student s**)**

**{**

**return** name**;**

**}**

char Student**::**getGender**(**Student s**)**

**{**

**return** gender**;**

**}**

string Student**::**getMajor**(**Student s**)**

**{**

**return** major**;**

**}**

int Student**::**getBirthyear**(**Student s**)**

**{**

**return** birthyear**;**

**}**

int Student**::**getBirthmonth**(**Student s**)**

**{**

**return** birthmonth**;**

**}**

void Student**::**setNumber**(**int number**)**

**{**

**this->**number **=** number**;**

**}**

void Student**::**setName**(**string name**)**

**{**

**this->**name **=** name**;**

**}**

void Student**::**setGender**(**char gender**)**

**{**

**this->**gender **=** gender**;**

**}**

void Student**::**setMajor**(**string major**)**

**{**

**this->**major **=** major**;**

**}**

void Student**::**setBirthyear**(**int birthyear**)**

**{**

**this->**birthyear **=** birthyear**;**

**}**

void Student**::**setBirthmonth**(**int birthmonth**)**

**{**

**this->**birthmonth **=** birthmonth**;**

**}**

LinkList中重载Locate两个函数：

//重载Locate函数

template **<**class T**>**

int LinkList**<**T**>::**Locate**(**int item**)**

**{**

Node**<**T**>\*** p **=** head**->**next**;**

int j **=** 1**;**

**while(**p **&&** p**->**data **!=** item**)**

**{**

p **=** p**->**next**;**

j**++;**

**}**

**if(**p**)**

**return** j**;**

**else**

**return** 0**;**

**}**

编译错误：



