数据结构课程设计

项目说明文档

考试报名系统

软件工程

张靖凯

2151396

目录

项目要求

项目简介

考试报名工作给各高校报名工作带来了新的挑战，给教务管理部门增加了很大的工作量。本项目是对考试报名管理的简单模拟，用控制台选项的选择方式完成下列功能：输入考生信息；输出考生信息；查询考生信息；添加考生信息；修改考生信息；删除考生信息。

项目功能要求

本项目的实质是完成对考生信息的建立，查找，插入，修改，删除等功能。其中考生信息包括准考证号，姓名，性别，年龄和报考类别等信息。项目在设计时应首先确定系统的数据结构，定义类的成员变量和成员函数；然后实现各成员函数以完成对数据操作的相应功能；最后完成主函数以验证各个成员函数的功能并得到运行结果。（建议采用链表实现）

系统示例

项目设计

数据结构设计

使用配有迭代器的双向循环链表来实现考试报名系统数据的增删改查，时间复杂度低，效率高。

插入	删除	访问
push_front O(1)	pop_front O(1)	O(n)
push_back O(1)	pop_back O(1)	
insert O(1)	erase O(1)	

类设计说明

命名空间 MercedesKK 中共三个类，另有本题中的 Student 类。

N MercedesKK	
C List	仿STL中的List
C ListIterator	用迭代器封装指针，重载++ – ()等operator 同时迭代器种类设置为bidirectional_iterator_tag
C Node	结点类

Node 类

Node 的对象为 list 中的每个节点，其中有成员 value_type 类型的数据和 NodePtr 类型的前后指针，默认构造函数初始化均指向自身。

Public Types

using	value_type = T
using	NodePtr = Node< T > *
using	NodeRef = Node< T > &

Public Attributes

value_type	_data 元素 More...
NodePtr	_next 后指针 More...
NodePtr	_prev 前指针 More...

ListIterator 类

ListIterator 仿照 STL，是 List 配有的迭代器，其中为了满足 traits 特性，将类型别名规范；同时设置 iterator category 为 bidirectional_iterator_tag 类型；封装指针过程中重载了自增自减等运算符，同时还搭配了 const 类型的重载。

Public Types

using	**LinkNode** = **Node**< T >
using	**self** = **ListIterator**< T, Ref, Ptr >
using	**iterator_category** = std::bidirectional_iterator_tag
using	**value_type** = T
using	**pointer** = Ptr
using	**reference** = Ref
using	**size_type** = size_t
using	**difference_type** = ptrdiff_t
using	**const_reference** = const Ref
using	**const_pointer** = const Ptr

Public Member Functions

	ListIterator (**LinkNode** *pnode=nullptr) 指针转迭代器构造函数 More...
	ListIterator (const **self** &lt) 拷贝构造函数 More...
reference	**operator*** () 重载* More...
pointer	**operator->** () 重载-> More...
const_reference	**operator*** () const 重载* const类型 More...
const_pointer	**operator->** () const 重载-> const类型 More...
bool	**operator!=** (const **self** &x) const 重载!= More...
bool	**operator==** (const **self** &x) const 重载== More...
self &	**operator++** () 重载++ More...
self	**operator++** (int) 重载++ More...
self &	**operator--** () 重载-- More...
self	**operator--** (int) 重载-- More...

List 类

本类仿照 STL 中的 list，为双向循环链表。同样使用了标准 STL 中的类型别名。实现了 const 和非 const 迭代器。构造函数有迭代器区间构造函数，构造指定 n 个元素函数,拷贝构造函数。重载了多个 insert 函数，包括插入一个数据，多个数据，通过迭代器插入数据等等，erase 函数同理。也提供了同 STL 中的 push_back 等函数，在内部实现是统一使用 insert 进行处理。

Public Types

using **value_type** = T

using **difference_type** = ptrdiff_t

using **size_type** = size_t

using **pointer** = **value_type** *

using **reference** = **value_type** &

using **LinkNode** = **Node**< T >

using **iterator** = **ListIterator**< T, T &, T * >

using **const_iterator** = **ListIterator**< T, const T &, const T * >

Public Member Functions

iterator **begin** () noexcept
头迭代器 More...

iterator **end** () noexcept
尾迭代器 More...

const_iterator **cbegin** () const noexcept
const 头迭代器 More...

const_iterator **cend** () const noexcept
const 尾迭代器 More...

List< T > & **operator=** (**List**< T > &lt)
赋值运算符重载 More...

void **push_back** (const T &val)
尾插 More...

void **push_front** (const T &val)
头插 More...

void **pop_back** ()
尾删 More...

void **pop_front** ()
头删 More...

iterator **insert** (**iterator** pos, const T &val=T())
任意位置插入一个数据 More...

void **insert** (**iterator** pos, int n, const T &val)
任意位置插入多个数据 More...

template<class InputIterator >
void **insert** (**iterator** pos, InputIterator first, InputIterator last)
任意位置插入一个迭代器区间的数据 More...

iterator **erase** (**iterator** pos)
任意位置删除一个数据 More...

iterator **erase** (**iterator** first, **iterator** last)
删除一个迭代器区间的数据[first, last) More...

iterator **find** (**iterator** first, **iterator** last, T element)
查找元素 More...

int **find** (const T &element)
查找元素 More...

bool **empty** ()
判空 More...

size_t **size** ()
计算个数 More...

void **clear** ()
清除 More...

void **swap** (**List**< T > &lt)
交换 More...

List () noexcept
无参构造函数 More...

template<class InputIterator >
List (InputIterator first, InputIterator last) noexcept
迭代器区间构造函数 More...

List (int n, const T &val=T()) noexcept
构造指定n个元素函数 More...

List (const **List**< T > &lt) noexcept
拷贝构造函数 More...

~List ()
析构函数 More...

Private Attributes

LinkNode * **_head**
头指针 More...

Student 类

Public Member Functions

	Student () constructor More...
bool	**operator==** (const **Student** &s)

Public Attributes

int	**number** 考号 More...
std::string	**name** 学生名字 More...
std::string	**gender** 性别 More...
int	**age** 年龄 More...
std::string	**category** 报考种类 More...

设计亮点

代码注释规范

采用了 doxygen 注释规范，对类、函数等有简要说明，命名采用驼峰命名法，类内的各个声明规范，方便本人回顾之前写过的代码和 code reviewer 查看，使 API 规范，帮助这个开发流程高效、规范地进行。

```
/**
* @class ListIterator
* @brief 用迭代器封装指针，重载++ -- ()等operator
* 同时迭代器种类设置为bidirectional_iterator_tag
*/
template<class T, class Ref, class Ptr>
class ListIterator
{
public:
    using LinkNode          = Node<T>;
    using self              = ListIterator<T, Ref, Ptr>;

    using iterator_category = std::bidirectional_iterator_tag;
    using value_type        = T;
    using pointer           = Ptr;
    using reference         = Ref;
    using size_type         = size_t;
    using difference_type   = ptrdiff_t;
```

```
public:
    iterator begin()       noexcept { return _head->_next; }          ///< 头迭代器
    iterator end()         noexcept { return _head; }                 ///< 尾迭代器
    const_iterator cbegin()const noexcept { return _head->_next; }    ///< const 头迭代器
    const_iterator cend()  const noexcept { return _head; }           ///< const 尾迭代器
```

输入错误处理

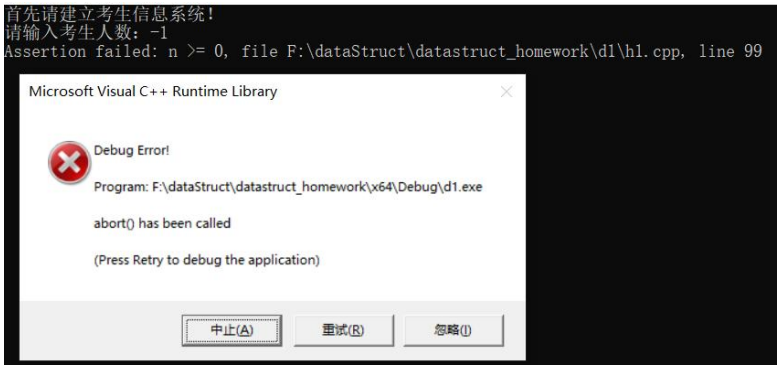
```
while (1)
{
    std::cin >> student.number >> student.name >> student.gender >> student.age >> student.category;
    if (student.number < 0 || student.number > 0x3f3f3f3f || student.age < 0 || student.age > 150 || !std::cin)
    {
        std::cin.clear();
        std::cin.ignore(1024, '\n');
        std::cerr << "输入考号/年龄不合理，请重新输入" << std::endl;
        continue;
    }
    break;
}
```

```
首先请建立考生信息系统！
请输入考生人数：2
请依次输入考生的考号、姓名、性别、年龄及报考类别！
2151396 张靖凯 男 20 软件工程
2151377 魏嘉明 男 20 工业设计

考号            姓名            性别            年龄            报考类别
2151396         张靖凯          男              20              软件工程
2151377         魏嘉明          男              20              工业设计


请选择您要进行的操作（1为插入，2为删除，3为查找，4为修改，5为统计，0为取消操作
1
请输入你要插入的考生的位置：3
请依次输入要插入的考生的考号，姓名，性别，年龄及报考类别！
xx!2xx 张翔 男 19 软件工程
输入考号/年龄不合理，请重新输入
```

断言处理

当输入学生人数不合常理即直接断言错误退出程序。

```
int n;      ///< student number
std::cin >> n;
assert(n >= 0), assert(n < 0x3f3f3f3f);
```



边界处理完善

如不可以插入学号相同的学生信息。（更多可见 exe）

```
考号            姓名            性别            年龄            报考类别
1               1               1               1               1
2               2               2               2               2


请选择您要进行的操作（1为插入，2为删除，3为查找，4为修改，5为统计，0为取消操作
1
请输入你要插入的考生的位置：1
请依次输入要插入的考生的考号，姓名，性别，年龄及报考类别！
1 2 3 4 5
考号重复！
```

Linux 测试（Ubuntu）

```
kk@LAPTOP-UJDPHKT8: ~/da
 1 #pragma  execution_character_set("gbk")
 2 #include "List.hpp"
 3 #include <iostream>
 4 #include <string>
 5 #include <iomanip>
 6 #include <cassert>
 7
 8 using namespace MercedesKK;
 9
10 static const std::string select_string = "请选择您要进行的操作（1为插入，2为删除，3为查找，4为修改，5为统计，0为取消
   操作";
11
12
13 /// @brief 在当前作业中设定的学生类
14 class Student
15 {
16 public:
17     int number;             ///< 考号
18     std::string name;       ///< 学生名字
19     std::string gender;     ///< 性别
20     int age;                ///< 年龄
21     std::string category;   ///< 报考种类
22
23     /// constructor
24     Student() : number(-1), name(""), gender(""), age(-1), category("") {}
25
26     bool operator==(const Student& s)
h1.cpp                                                          1,1           Top
"h1.cpp" 221L, 6956C
```

```
kk@LAPTOP-UJDPHKT8:~/dataStruct/homework1$ vim h1.cpp
kk@LAPTOP-UJDPHKT8:~/dataStruct/homework1$ ./run
首先请建立考生信息系统！
请输入考生人数：2
请依次输入考生的考号、姓名、性别、年龄及报考类别！
2151396 张靖凯 男 20 软件工程
1234567 呜呜呜 男 20 计算机科学与技术

考号            姓名            性别            年龄            报考类别
2151396           张靖凯          男              20                软件工程
1234567           呜呜呜          男              20                计算机科学与技术


请选择您要进行的操作（1为插入，2为删除，3为查找，4为修改，5为统计，0为取消操作
1
请输入你要插入的考生的位置：1
请依次输入要插入的考生的考号，姓名，性别，年龄及报考类别！
7654321 某某某 男 20 大数据

考号            姓名            性别            年龄            报考类别
7654321           ◆某某某         男              20                大数据
2151396           张靖凯          男              20                软件工程
1234567           呜呜呜          男              20                计算机科学与技术


请选择您要进行的操作（1为插入，2为删除，3为查找，4为修改，5为统计，0为取消操作
```