**C++第三次（期末）大作业**

**提交日期：**2025年1月3日晚12:00之前

**要求：**建立1个项目，包含3个子目录，各提交一道大题的程序代码，以及各道题运行结果的文本文件result.txt，发送给助教，邮箱同前几次。

**理论课题目**

1. Club Center
2. 设计Person类，包含std::string id\_, std::string name\_;
3. 设计Student类，继承自Person，额外包含std::string schoolName\_, double discountRate\_；
4. 设计Date类，包含int year\_, int month\_, int day\_；
5. 设计Activity类，包含Date date\_, std::string place\_, std::string activity\_, std::set<Person\*> members\_;
6. 设计Club抽象基类，包含std::string name\_, std::set<Person\*> members\_; std::set<Activity\*> activities\_ ，纯虚函数displayMembers()，displayActivities()，addMember(const Person&), addActivity(const Activity&);
7. 设计SportsClub类，继承自Club，额外包含枚举变量interest\_ (比如Running, Swimming)，std::string coach\_；重写纯虚函数。
8. 设计MusicClub类，继承自Club，额外包含枚举变量instrument\_ (比如Piano, Violine等)，重写纯虚函数。
9. 设计ClubCenter类，包含std::vector<Club\*> clubs\_，提供函数添加Club对象，并遍历各个club，打印成员和活动信息。

提供main函数，测试各项功能。手工建立4个Club对象，两个SportsClub，两个MusicClub，各包含1个Person和2个Student对象，加入到ClubCenter对象当中。注意深拷贝保证内存安全，不要随意引用栈上的对象（临时对象）。可以考虑将原始指针改成智能指针。

1. ListQueue<T >
2. 把第一次大作业的pointvector作业拓展成模板实现，实现MyVector类模板。实现基本数据类型如int, double,以及自定义类型Student（第一题）的实例化。
3. 由于MyVector的头部和中间增删开销太高，实现一个ListQueue类模板，如下图所示。利用双向链表（自己实现，不能直接用stl::list）组织多个MyVector的对象，每个MyVector对象的长度上限可以事先规定（如100）；当超过上限时需要创建新的MyVector对象，并在双向链表里面做相应节点插入；当一个MyVector对象被清除时，也需要在双向链表里面删除相应节点。一个ListQueue对象的接口类似stl::deque，除了std::vector的基本API外，还提供push\_front, pop\_front, pop\_back, front，insert, erase，以及随机访问（[]）等功能。最后，提供一个print函数，展示内部状态（当前的节点个数，各个MyVector的元素个数）。
4. 提供测试程序，以ListQueue<int>为例完成所有接口API的测试，不能遗漏。另外，还要求如下测试：

* 在当前的容器中，以[]的方式随机访问一个元素，然后在它之前插入一个新元素，直到容器中达到100K个元素
* 对元素进行随机删除，直至ListQueue对象为空。

双向链表

MyVector

MyVector

MyVector

**实验课题目**

本次作业模拟数据库的操作，主要读取StudentInfo.txt和StudentCourse.txt两张表，其中StudentInfo.txt 为学生个人信息表，包括学号id、姓名name、性别sex、出生年月birthday、学年schoolYear、籍贯birthplace，StudentCourse.txt为学生选课信息表，包括学号id、选课course、学分credits、分数score。

1. 读入两个文本文件StudentInfo.txt和StudentCourse.txt，使用合适的STL数据结构存储两张表的信息，要求以下的操作效率要高，不建议临时做线性遍历（即n个元素逐一访问）。查询结果需要联表打印出相关信息；如果有多个条目，默认按照id排序。
2. 提供排序操作，根据各个字段进行排序，比如SortByName, SortByScore（单项课程分数），SortByTotalScore（总分数），支持范围查询，比如查询某门课分数在80-90之间的学生，注意合理设计接口，尽量做到代码简洁，少冗余。
3. 提供查询和删除操作，根据各个字段进行条件删除。支持多种条件比如sex == ‘M’ && (birthday.year > 2019 || score < 80)（不用解析这个字符串表达式），注意删除要保持两个表格的一致性。
4. 提供Test函数，具体测试以下用例，将结果**通过ofstream写入**到result.txt中
   1. 打印2020级选修C语言且成绩小于60分的学生
   2. 统计课程平均分大于80的学生个人信息并输出
   3. 查询每个学生是否修满20学分

附：

将以下文本保存至StudentCourse.txt

#id course credits score

10917 C Programming Language 3 73

10919 C Programming Language 3 91

10914 C Programming Language 3 59

10908 C Programming Language 3 56

10921 C Programming Language 3 83

10905 C Programming Language 3 74

10917 Advanced math 4 77

10919 Advanced math 4 76

10914 Advanced math 4 70

10908 Advanced math 4 80

10921 Advanced math 4 79

10905 Advanced math 4 53

10917 College English 3.5 90

10919 College English 3.5 83

10914 College English 3.5 79

10908 College English 3.5 80

10921 College English 3.5 81

10905 College English 3.5 70

10917 Control Theory 2 88

10919 Control Theory 2 86

10914 Control Theory 2 87

10908 Control Theory 2 85

10921 Control Theory 2 90

10905 Control Theory 2 91

10917 Python 2 72

10919 Python 2 81

10914 Python 2 69

10908 Python 2 96

将以下文本保存至StudentInfo.txt

#id name sex birthday schoolYear birthplace

10905 xiaomei Female 1999/2/1 2021 hunan

10908 jianfa Male 1999/7/2 2020 hunan

10914 dahong Male 2000/2/3 2021 guangdong

10917 xingfeng Male 2000/2/4 2020 jiangsu

10919 xiaomi Female 2001/12/5 2019 beijing

10921 dabin Male 1998/2/6 2018 hainan