# “语言课程设计”报告

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 姓 名 | | 王亮 | 程 序 | 满分 优 良 中 及格 不及格 |
| 学 号 | | JS320330 | 文 档 | 满分 优 良 中 及格 不及格 |
| 专 业 | | 计算机类 | 平 时 | 满分 优 良 中 及格 不及格 |
| 起始日期 | | 2021 年 7月 | 总 评 |  |
| 终止日期 | | 2021 年 7月 |
| 设 计 文 档 | | | | |
| 设计任务 | 题目一 银行定期储蓄管理系统   1. 创建存款文件(第一次输入储户信息时)； 2. 创建取款文件(第一次办理取款时)； 3. 接收储蓄。接收储户与营业员信息并将以上信息添加到存款文件中； 4. 处理挂失。根据储户提供的帐号、姓名、密码、地址、储种，储金核查有无此项存款，有则对此帐号加入挂失标记，若无此项存款，则需判断是否领走还是未发生过这笔存款。 5. 办理取款：(1)根据储户提供的存款单(上面有帐号、姓名、储种、本金、日期)判断是否到期，检查有否挂失，根据储户提供的密码判断是否正确。若以上判断通过，执行以下三步。(2)在取款文件中添加这笔取款的储户和营业员的所有信息和取款日期，以便复查。(3)打印利息单，其内容包括：帐号、姓名、储种、起息日期，支取日期、本金、利息和支取金额。利息计算方法如下：1年到期利率1.98％；3年到期利率2.25％：5年到期利率３.5％，每逾期一天，每天利率0．05‰；若未到期提前支取，按每天利率0．05‰计,利息税按20%扣除。(4)在存款文件中对这笔存款删除。 6. 统计今后3天每天到期的储金数，以便备款。 | | | |
| 任务分析 | 必须任务分析：   1. 程序定位：面向银行业务员的业务管理系统 2. 程序核心功能：增删改查CRUD（更改由删除再添加实现），通过显示屏显示内容进行可视化交互。 3. 设定：    1. 账号原则上作为“唯一身份识别”，以处理同名(同一人)多笔存款，与同名(非同一人)单笔\多笔存款；    2. 原则上单次使用(每一次打开应用)业务员是固定的(对应实际情景中：共有多个业务窗口，其中单个窗口业务员在一段时间内为同一人处理各项业务。但尚未掌握网络编程技术，未能实现：1.不同窗口业务员进行通信2.换班(非本地))。作为代替，在本地上可以“更换业务员”进行更换(默认的业务员)，可以“指名业务员”进行指定更换，若不存在则当即创建并录入信息。    3. 实际业务中有根据身份证号、人脸信息等获取认证信息的功能，程序中使用“一键填入”进行代替，为方便调试，密码也进行自行填入。(取款日期默认到期，可以自行调整)    4. 文件保存分为三类：默认保存（保存至默认文件路径（使用相对路径，在debug文件夹下，如下图）），另存为（自选路径），保存至上一次路径。应自行选择是否保存。   若为使用windeployqt打包好的程序则在程序根目录下   1. 创建存款文件    1. 需要构造存款类(savings)作为操作对象，一条存款为一个存款类(savings)对象，存款应当包括：储户信息与营业员信息。其中储户包含账号、姓名、密码、地址、储种、本金收储、日期、是否挂失、挂失日期信息，应构造一储户对象(depositor类)与含营业员姓名与工号信息的营业员对象(clerk类)。存款文件用于保存当前所有的存款信息(其中包括丢失的，不包括领走的)，每次创建新的取款(savings)对象时，写入存款文件(如果没有此文件，会自行创建，与第一次输入创建文件契合)。 2. 创建取款文件 3. 需要构造取款类(Withdrawl)作为操作对象，一条取款为一个取款类(Withdrawl)对象，其中应包括从存款中可获取的信息(取款时存款中信息应保留下来)，所以可以以存款(savings)做为其组成成员之一，此外，它还具有支取日期，利息，支取金额的额外信息，作为剩余的组成成员。在文件操作上，题目中没有提及除第一次办理外的操作，没有提及是否删除存款（打印利息单后），所以仅首次创建时自动保存，其他时候应使用提供的保存功能保存存款与取款，此外，取款创建后不自动删除，用户可以点击显示屏中对于取款，手动删除。  1. 接收储蓄 2. 允许“新增存款(savings)”功能新建一存款对象保存储户与营业员信息。 3. 处理挂失 4. 允许“挂失处理”功能根据储户提供的账号、姓名、密码、地址、储种、储金进行做以下三件事：1.有无此存款(遍历寻找) 2.找到加入挂失标记(修改找到的对象属性—是否挂失与挂失日期，提供的信息应补上提供挂失日期) 3.刷新显示 5. 办理取款 6. 存款单填入信息后进行：1.遍历寻找判断是否挂失 2.判断寻找到的对象密码与输入是否一致 3.计算时间差，获得利息与取款金额 4.删除并添加到 5.打印利息单 7. 统计到期储金 8. 计算当前日期（用户进行输入）与每一笔存款到息日期之差（若当前日期大于到息日期不计），查找差值小于等于2天的存款，即距到息日期：0天，1天，2天，则显示以下信息于窗口中：距离天数、总共到期储金数、总共应付储金额，并将主窗口显示屏中对应存款标红，退出后仍保留标红便于对照，双击存款并退出详情窗口可以取消标红。 | | | |
| 设计思路与流程 | 1. 设计基本的、核心的操作对象——存款(savings)与取款(Withdrawl)类，包含相关信息作为成员变量。可以提供对外public接口或是通过友元类进行访问成员； 2. 题目中提到存款(savings)类中应包含：1.储户信息(depositor)类 2.营业员信息(clerk)类 故设计此二类 作为savings类成员； 3. 使用营业员信息集(clerkinfo)类对象打包所有营业员(默认数据)； 4. 进行初次界面设计，进一步探究对象间关系。具体：使用Qt Designer：1.大致设计主界面mainwindow.ui；2.功能界面-挂失处理dialoglosthandler.ui 3.功能界面-取款处理dialogwithdrawls.ui 4.新建存款界面dialog.ui 5.打印利息单窗口dialoglist.ui； 5. 对于主页面，通过加入一系列QAction连接到各功能——挂失处理 取款处理 新建存款 如下图所示：  1. 在主窗口通过connect将对应QAction与对应窗口创建与具体处理联系起来   其中以下功能包含：  窗口间关系：   1. 新建存款是会调用保存功能中保存存款(创建文件并保存)的 2. 取款处理的第一次会调用保存功能中保存取款(创建文件并保存)的 3. 办理取款后自动调用打印利息单   安全性问题：(均提供提示，可修改已填入信息)   1. 办理取款不可办理已挂失存款 2. 办理取款对密码错误的不办理 3. 挂失处理对不存在存款不办理 4. 拒绝空白输入   完成功能函数(部分使用lamda表达式)：   1. 计算利息与应取金额（逾期，未到期，到期） 2. 打印存款单（办理取款的调用，用户使用的调用） 3. 取款窗口判断日期合法性（原则上不得逾期超过三年，不得早于存款日期，挂失窗口不加限制） 4. 查找将到期存款   ————————以下为非基本要求设计流程————————   1. 在主窗口提供排序功能，便于查找 2. 考虑使用sortItem的接口，但并不能直接自写谓词； 3. 最终使用Qsort对容纳存款的qlist与对应取款的qlist进行排序后刷新显示 例如下：  1. 因为QList中未能找到反转链表的接口，也没看到有qReverse，使用了STL中的std::reverse进行反转，用于改变升序与降序。  1. 允许记录此时排序序列，可通过回退功能回退至上次记录   注：此功能是一种残缺的替代，希望实现的是进行多个标准组合排序(如在本金降序前提下，进行姓名升序，再进而进行到息日期降序等以此组合下去)，在此基础上能进行单步回退。实际实现上，除了记录每次动作的list内容，暂不了解其他方法，能力有限，谨以此替代。   1. 新增项提供标识，退回、读取回到过往状态失去标识   创建新增项时ListWidgetItem提供图片，读取已创建项时刷新窗口，重新创建的ListWidgetItem不带图片   1. 提供存取款读取功能   QIODevice 读取文件操作   1. 输入补全功能 2. 更换和指名工作人员   读取linetext内容创建clerk对象(QString方式，初期使用QStringList)。   1. 播放音乐   使用QMediaPlayer加歌单，QMediaplay进行播放，如果出现  解释如下：  参考：https://github.com/qyvlik/Flat.qml/issues/4  并且确保已添加multimedia | | | |
| 功能介绍 | * 总体：增删改查   基本功能：   1. 创建存款文件 2. 创建取款文件 3. 保存存款取款 4. 读取存款取款文件 5. 显示存款取款内容 6. 删除存款取款 7. 查看存取款详细 8. 挂失处理 9. 办理取款 10. 查询到期 11. 打印利息单  * 额外：   并不成功的拓展功能：   1. 详情显示（使用过QString arg函数，换等宽字体，也没成功实现文字对齐，此功能作为补偿） 2. 排序（单标准） 3. 序列保存与回退 4. 工作人员更换与添加 5. 夹带私货的音乐播放（点击播放音乐播放，再点击暂停） 6. 打印利息单（复查，对于所有取款可以再次打印回执单，输入账号点击打印即可） 7. 补全功能，默认最顶上的元素补全，输入账号则补全当前账号其他内容，为便于调试，补全了密码，默认补全日期为取款日期 | | | |
| 文件与对象描述 | 从上至下：   1. Clerk 营业员类 包含营业员基本信息 对外接口 2. Clerkinfo 营业员总和类，包含营业员全体信息 对外接口 3. Depositor 储户对象类 包含储户基本信息 对外提供信息获取接口 4. Dialog 新建存款窗口类 完成对用户窗口界面与后台操作 5. Dialoglist存款单类 同上 6. Dialoglosthandler 挂失处理 同上 7. Dialogwithdrawls办理取款 同上 8. Dialogwithdraltotake 查询 同上 9. Mainwindow 主界面窗口 关联各个窗口 将所有数据汇集统合 10. Savings 存款 核心类 11. Withdrawls 取款 核心类   Ui：上述所有窗口均使用ui进行设计界面，后台操作纯代码实现，不通过designer进行连接（connect）响应  只使用了部分的资源文件，提交代码时由于邮件附件限制(30M+ )可能不会上传音乐。 | | | |
| 测试报告 | 功能流程：  (建议复制流程文本到新txt按流程对照看图)新建存款->文件创建->取款办理->文件创建->接受储蓄->补全信息->挂失办理->查看详细->删除->排序->读取文件->利息单复查->查询到期->另存为->更换工作人员->指名工作人员(不在列表当即创建)->更换(验证创建)->退出询问： | | | |
| 总结 | 不足：   1. 缺乏优秀的交互界面设计 2. 缺乏互动反馈(feedback)  * 按键没有交互动画(并没有找到合适的素材) * 按键与界面弹出响应没有配上适宜的交互音效（只有将就用的）  1. 界面设计过于简陋  * 主页面，各功能页面没有足够优秀的背景设计（素材问题）  1. 数据结构的选用 2. 查找的频繁  * 尽管提示中选用了链表，但对于如此频繁的查找(检查是否挂失，是否密码正确，以及其他一些安全性问题，这些都提出定位目标元素的需求)，链表是不便的。是需要一个个遍历查找，每做一次此类检查，都需要遍历一次，尽管使用的测试数据也没多少，看不出速度区别，但真正设计系统时，O(n)的复杂度是完全不行的。不妨添加每个用户的唯一标识，采用哈希表。事实上，在后续处理中我已经将账号当做客户唯一标识，以避免出现同名下一人多号与多人一号混淆。这样看来，尽管已经使用链表写成程序，但确实不如一开始使用哈希表。  1. 删除操作  * 大体同上，删除需要先找到相应元素，整个程序都在频繁CRUD。   小结 && 心得：   1. 能够感受到一个完整的程序\产品设计的流程与思想 2. 更深入体会了面向对象 3. 感受到实战项目带来的提升   具体说：  对象间关系探寻是面向对象的重要一环，这之间联系未必是完全清晰的，对此的认识是在程序编写中动态更新的。刚上手立项，其实并不清楚如何组织起这些对象，至少从文字要求上看了也并不能完全“体会”到这之间的联系，信息是全面但又有缺漏的。(题目中没有提到：1.取款的后续保存以及删除 2.营业员新建？营业员保存？3.是否复利？(后续群中解答了这一点) 4.对于非法输入的处理(程序中采用正则表达式进行限制，但会由此无法使用中文，若使用中文，则无法使用正则表达式进行限制，考虑到平日使用的经验，仍选择正则表达式) 5.保存文件的读取？)于是先进行的是各个对象的构建，一开始没觉得有必要使用get函数，对象间关系是简单的，获取元素处理也是简单的，使用友元类，对于获取单个元素使用了QString中split函数进行处理，对应上文件操作与QlistWidget的显示。但后来随着外界获取类内元素需求增加，还有各个类中获取其组成元素的需求，get函数显得方便，也更具有可读性。体会到其必要性。建立完各个对象后，在聚焦于某个具体功能设计时，能感受到对象间的关系逐渐变得清晰起来(start to dawn on me)。  在实际项目中，感受到产品完整开发应有的协调合作，后台程序功能的实现跟前端面向用户的界面设计，是要相当程度协调匹配的。切实感受到音乐制作、动画设计在产品中重要的地位。  从零开始做一次实际的项目实在是受益匪浅，在实战项目中综合的知识应用给人提升很大。不如这么说，实战项目对Programmer提出各式各样的要求，而在做成某项需求的过程中，需要去不断掌握新的、不曾熟悉的知识(Qt界面设计(designer 各种widget 各种button edit控件) 多媒体(multimedia)等) 同时复习旧的知识，甚至有着更深的理解，温故而知新，感受到过往掌握的知识串了起来，有着更为深层的联系。Lamda表达式捕获模式 =与&，修改指针时指针的传递，正则表达式常用的通配符 此等。  随着程序设计进一步深入，更为切实感受到Programming的魅力所在，实现种种创意idea、感受专注concentration投入带来的独特体验、从零开始的创造creation、好奇curiosity驱使的学习进步，给生活带来十足的乐趣。不过平心而论，Qt里还有很多强大的功能并没有去使用，一些接口尚有些许疑惑，也许以后逛github gitee看到有意思的小程序能进一步学习掌握。  建议：建议换个邮箱收作业，公告要求总大小小于100MB，实际上东大邮箱甚至不允许我提交超过36MB的附件 | | | |

报告填写说明：

1. 设计任务：填写课程设计的任务及要求；
2. 任务分析：填写对设计任务的理解和规划；
3. 设计思路和流程：填写软件设计的基本思路和重要流程图；
4. 功能介绍：填写系统的主要功能；
5. 文件与对象描述：对系统的组成文件及其类对象进行介绍；
6. 测试报告：对所使用的测试数据及测试结果进行介绍，请在文中插入重要的软件的运行界面及数据显示结果图片（提示，图片截屏请使用快捷键Alt+PrintScreen或PrintScreen）；
7. 总结：本课程设计的小结、心得及建议。