

37-作业报告

（一）功能介绍

本项目编写的是一个基于 c++ 的记账程序，参考当前手机中常见的账本软件，能够实现以下功能：

- 1、记录账目收支，记录内容包括收支金额、记账时间（精确到日）、收支所属分类以及自定义文字备注；
- 2、查看所有记账条目，并可对账目的金额、备注进行修改；
- 3、通过时间索引查找记账条目，并可对账目的金额、备注进行修改；
- 4、自定义时间范围生成各类账目标签收支占比饼图

本项目分为以下几个界面：

- 1、开始界面，记账界面的大入口，能够显示总收入、总支出和结余，开始界面设有四个按钮，分别为“添加一条新记账”、“全部历史收支”、“按时间查找”、“记账数据统计”；

- 2、添加界面，在开始界面单击“添加一条新记账”按钮进入，分为收入和支出两块，在各自的面板中，均可选择账目类别、自定义金额、时间和备注；

- 3、全部历史收支的显示界面，在开始界面单击“全部历史收支”按钮进入，该界面会按照距今时间顺序从近到远显示历史收支条目，具体内容包括条目分类和该条目金额，可单击每条收支条目的“查看”按钮显示详细信息；

- 4、日历界面，在开始界面单击“按时间查找”按钮进入，该界面主体为日历，单击日历上的某一日期可在下方显示出当天的收入和支出，双击则可进入当天收支的显示界面；

- 5、统计界面，在开始界面单击“记账数据统计”按钮进入，在该界面中，可以自定义查询时间段的起始日期和结束日期，该界面会根据所选择的起止日期分别生成收入和支出的饼图，饼图上可以体现出各种类型收支的占比；

- 6、单日收支显示界面，在日历界面双击某一日期进入，该界面会显示当日的收支条目，具体内容包括条目分类和该条目金额，可单击每条收支条目的“查看”按钮显示详细信息，同时，可通过更改该界面上部的日期来更换查看的日期；

- 7、单条账目详细信息显示界面，可通过单击全部历史收支显示界面或单日收支显示界面中某条收支条目的“查看”按钮进入，该界面能够显示该条账目所属日期、类别以及账目金额和备注，其中，金额和备注是可以修改的，单击界面右下角的“保存”或“删除”按钮可分别实现保存修改和删除该账目的功能。

（二）模块细节

本项目分为以下几个模块：“data”，“mainwindow”，“addone”，“calendar”，“view_all_data”，“statistics”，“viewdata”，“showonedata”，“detail_of_data”，每个模块的实现细节如下：

- 1、“data”模块，主要定义了本项目中需要用到的一些全局变量和结构体，如：

double 类型的 income_all 和 outcome_all，用于记录总收入和总支出；
int 类型的 rank_now，用于记录当前条目的次序；

结构体 data_inf，用于记录一条账目的时间及其被记录的顺序，其中包

含 year, month, day, rank 几个 int 型的成员变量，同时还重载了<运算符，用于排序；

结构体 bill_one，用于记录一条账目的所有信息，该结构体利用 data_inf 型的 bill_data 记录日期、次序，string 型的 name 记录类别，bool 型的 isincome 记录是否为收入，double 型的 money 记录金额，QString 型的 remark 记录备注，同时也重载了<运算符，用于排序、储存 bill_one 型数据的 vector 容器 bill_all，用于储存所有账目信息。

2、“mainwindow”模块，实现读档和存档功能以及开始的 ui 界面；

开始时调用 loadDataFromFile 函数从“bill.txt”中读取存档，存档中每条账目的格式为：第一行为类别名字，第二行为金额，第三行为日期，第四行为备注，第五行为是否为收入（1 代表收入，0 代表支出）；

每读取一条账目信息使用该条信息初始化一个 bill_one 类型的 bill，同时更新 rank_now，并根据金额更新 income_all 和 outcome_all，然后将该 bill 插入 vector 容器 bill_all 中；

随后调用 updateLabels 函数更新开始界面的总收入、总支出和结余；

该模块中还定义了四个槽函数，其中，on_pushButton_clicked 函数实现了跳转到添加界面的功能，并在添加界面关闭后实时更新开始界面的相关数据；on_sta_clicked 函数实现了跳转到统计界面的功能；on_to_calendar_clicked 函数实现了跳转到日历界面的功能，并在日历界面关闭后实时更新开始界面的相关数据；on_all_history_pushButton_clicked 函数实现了跳转到全部收支显示界面的功能，并在该界面关闭后实时更新开始界面的相关数据；

最后，在该界面关闭前，会调用 closeEvent 函数向“bills.txt”中写入 bill_all 中的账目信息。

3、“addone”模块，实现创建一条新帐目的功能；

ui 界面主体是 tabWidget 类，分成收入和支出两栏。每一栏中都有具体类别（分为六个常用类和一个其它类），ui 界面还附上了相应的 icon。具体选择类别时，这些 Qradiobutton 被放在同一格 layout 中，从而能实现单选的功能。

此外，“addone”模块中还有记录金钱，时间与备注的部分，分别依托 QDoubleSpinBox, QDateEdit, QTextEdit 来实现。在保存按钮被按下后，这几处的对应值都会被储存：分别取他们的 value，date 与 toPlainText，再被放入 bill，最后暂时存在 bill_all 中。

4、“statistics”模块，实现呈现某一时间段内记账数据的功能；

此页面通过两个 QDateEdit 控件收集用户想要查询的起止日期，并通过 click 确定按钮激活槽函数，从容器中读取相应时间段内账目的数据，并进行分类加和、计算比例，再利用 QChart 进行饼图绘制，将绘制好的收入与支出饼图显示在 UI 界面预置的 QTabWidget 框内的两个 layout 中。

5、“calendar”模块，此模块主要实现从日历中便捷查看任意日期的收支情况，且在双击某个日期后能够跳转到该日期的账目细则（viewdata 页面）以进行进一步查看、修改或删除等操作。

本页 UI 界面主体为一个 QCalendarWidget 控件，日历上方用一个 QLabel 实时打印用户单击选择的日期；日历下方用两个 QLabel 实时打印用户单击选择的日期对应的收支数额。实现方式为单击日期激活日历控件的 clicked

槽函数，槽函数将得到的 QDate 用于设置日历上方 Label 的 QString 为该日日期，同时从容器中读取该日的账目数据，计算当日的总收入与总支出，分别打印在日历下方相应的 Label。同时，双击日历的某一天会激活日历控件的 activated 槽函数，该函数将会自动打开一个 viewdata 界面并设置 viewdata 界面的初始日期为用户双击的日期，以展现该日收支细目。

6、“viewdata”模块，实现显示某天所有收支的 ui 界面；

开始时通过 set_date_from_calendar 函数获取起始日期，随后也可通过 ui 界面上部 QDateEdit 类的 view_dateEdit 获取日期；

获取日期后，调用 on_OK_pushButton_clicked 函数，在 ui 界面中部 QListWidget 类的 view_listwidget 中显示账目，具体方法如下：首先对 bill_all 进行排序，随后遍历 bill_all 中的数据，若找到当天的账目信息，则 new 一个 QListWidgetItem，并用该 item 和 bill_all 中的这条账目初始化一个新的 showondata 类型的指针 tmp，并将其显示在 view_listwidget 上，若没找到，则显示“当日无收支”；on_OK_pushButton_clicked 函数中还通过 connect 函数将 item 与 showondata 中定义的 emit_del 函数联系起来，以确保 showondata 接收到“删除”信号时 viewdata 中实时删除 view_listwidget 中的对应 item；

7、“view_all_data”模块，实现显示所有历史收支的 ui 界面；

具体方法与 viewdata 类似，不同之处为，view_all_data 中采用反向迭代器来遍历 bill_all，从而使账目信息按照距今时间从近到远显示。

8、“showondata”模块，定义 viewdata 模块和 view_all_data 模块中 QListWidget 中的控件，显示单条账目；

在初始化 showondata 时，多传入了两个参数，一个是 QListWidgetItem 类型的指针 pitem，表示该 showondata 在 QListWidget 中所对应的位置，另一个是 bill_one 类型的指针 bill_to_show，表示该 showondata 要显示的数据属于哪条账目；

showondata 右部设有“查看”按钮，单击后调用 on_detail_pushButton_clicked 函数，跳转到单条账目的显示界面，当单条账目的显示界面关闭后，判断传入的 bill_to_show->money 是否为-1，若为-1，说明在单条账目的显示界面进行了删除操作，此时，先执行 emit_del 函数，将该条账目对应的 item 从所在的 QListWidget 中删除，随后通过 rank 和 type 在 bill_all 中找到该条账目，并将其移除 bill_all；若不为-1，说明没有进行删除操作，此时可能进行了修改操作，于是实时更新 ui 界面的金额。

9、“detail_of_data”模块，实现显示单条账目的具体信息的 ui 界面；

在初始化 detail_of_data 时，多传入了一个 bill_one 类型的指针 bill_to_show 表示要显示的数据属于哪条账目；

ui 界面采用 QLabel 来显示日期和类别，这两者无法修改，采用 QDoubleSpinBox 显示金额，QTextEdit 显示备注，这两者可以修改；

单击“保存”按钮调用 on_save_pushButton_clicked 函数，用此时 money_doubleSpinBox 和 remark_textEdit 中的内容替换原来的金额和备注，并同步对 income_all 或 outcome_all 做出修改；

单击“删除”按钮调用 on_delete_pushButton_clicked 函数，弹出对话框提示是否删除，若选择“yes”选项，则将 bill_to_show->money 的值改

为-1，表示此 bill_to_show 在 showonedata 中需要被从 bill_all 中移除，同时更新 income_all 或 outcome_all 的值，随后关闭界面。

（三）分工情况

本项目的小组分工如下：

陈泊霖负责“calendar”模块和“statistics”模块的实现，同时负责部分美工；

黄炫铭负责“data”模块、“mainwindow”模块和“addone”模块的实现，同时负责部分美工；

谢宇博负责“view_all_data”模块、“viewdata”模块、“showonedata”模块和“detail_of_data”模块的实现。

（四）总结反思

在这次的项目合作中，我们组对自身的整体表现较为满意。在合作过程中，我们能够较快地明确各成员的分工，按时完成自己的部分，团队氛围和谐，成员间始终保持沟通，确保各模块间能够密切结合，不存在影响项目进程的分歧，全体成员都任劳任怨地朝着共同的目标推进，最终成功地实现了设想中的各种功能，达到了预期的效果。

当然，由于经验的不足，我们组在合作过程中也遇到了许多问题，譬如因为时间规划不合理导致前期进程较赶、因为 qt 相关知识储备不足导致 ui 界面的实现过程磕磕绊绊等。不过，在全体成员的共同努力下，这些问题都得到了有效解决。

通过这次的项目合作，我们充分全面地掌握和巩固了本学期所学的知识，加深了对类的理解，熟悉了各种容器的使用，同时也学会了如何使用 qt 中的各种丰富的 ui 组件。qt 大作业极大地锻炼了我们的自学能力，遇到困难时，我们采用小组讨论、头脑风暴、在科技论坛查询等等多种方法，掌握了 qt 的大量特性与编程技巧，学习了 github 这一强大工具网站的使用，培养了较为复杂项目的项目进程管理与组织能力，程序的美工设计也提升了我们的计算机美术品味。在合作过程中，我们认识到了，要想完成一个项目，不仅需要扎实的知识储备和丰富的创新能力，还需要分工明确、协作默契的团队和合理的时间规划，如此才能确保项目有条不紊地推进。

相信通过此次 qt 大作业的合作，我们今后其他项目的合作一定会更加地得心应手，完成得更加出色。