

**《面向对象的程序设计》**

**大作业**

**题目：** 房贷计算器设计报告

物联网工程学院 **学院** 计算机科学与技术 **专业**

学 号： 1033190314

学生姓名： 张朱涵

班 级： 计科1903

成 绩：

**2020年 6月**

# 题目要求

* 1. 请使用QT或者其他类似的界面库设计人机交互界面，要求界面友好，方便使用。
  2. 请自行调研房贷的相关知识，并根据调研情况进行程序设计。
  3. 程序设计时必须使用面向对象的思想进行设计，使用C++编程。
  4. 实现功能需要包括商业贷款、公积金贷款和组合贷款的利息计算三种功能。

# 需求分析

随着生活水平的提高，人们买房的需求越来越大，在我们的日常生活中，经常会遇到房贷的计算，手工计算非常的繁琐，这时就需要用到一个专门的房贷计算器，所以借着课程设计这个机会我便开发了这个房贷计算器。计算器要实现的功能有以下几个方面：计算等额本息还款法的还款总额、总支付利息和月供，计算等额本金还款法的还款总额、总支付利息、首期付款、贷款月数、首月还款和每月换款递减数。

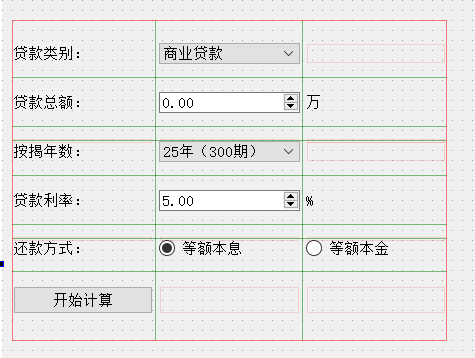
# 设计思路

* 1. 输入和输出

第一步确定程序的输入和输出。输入分为三类：商业贷款、公积金贷款和组合贷款。商业贷款的输入有贷款总额、按揭年数、贷款利率和还款方式，公积金贷款的输入与商业贷款相同，组合贷款的输入由商业贷款的输入和公积金贷款的输入组合而成。输出分为两类：等额本息和等额本金。等额本息的输出有每月月供、支付利息和还款总额，等额本金的输出有每月月供、每月递减、支付利息和还款总额。

* 1. **设计界面**

确定好输入输出后就可以设计界面了，商业贷款界面中包含以下控件：



radioButton\_2

doubleSpinBox

comboBox\_4

doubleSpinBox\_2

comboBox

pushbutton

Label\_11

Label\_9

Label\_8

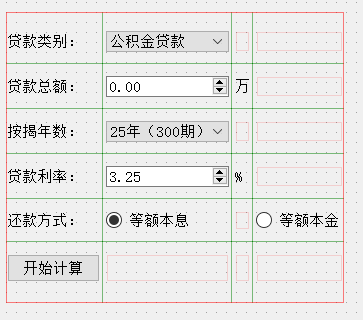
Label\_2

label

radioButton

其中，doubleSpinBox只能输入两位小数，并且设置了上下限，这使得程序不会因为用户错误的输入就崩溃，保证了程序的健壮性。

公积金贷款界面：



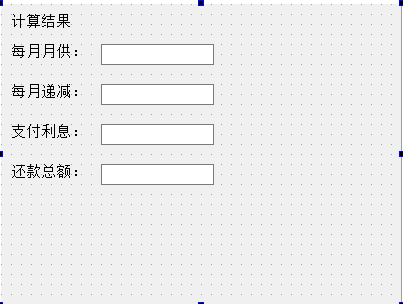
组合贷款界面：



等额本息界面：



等额本金界面：



输出框改为只读模式，这样用户就无法修改输出的内容。

* 1. **设计类**

界面设计完后再考虑如何设计类来实现计算器的各个功能。

以商业贷款为例，首先设置私有成员sum, year, rate和对应的成员函数实现对私有成员赋值和取值。页面之间的切换则是由信号和槽的机制来实现，槽函数（slots）用来处理信号。按下开始计算按钮后，*on\_pushButton\_clicked()*函数根据radio button的选项来显示计算结果界面；当贷款类别即comboBox中的内容改变时，函数*on\_comboBox\_currentIndexChanged(int index)*将触发信号*mixloanshow()*或*pfloanshow()*，对应的界面收到信号后，槽函数*receivesignal()*会显示该界面并将上一个界面隐藏。

接下来就是设计程序核心的功能：计算功能。由于商业贷款、公积金贷款和组合贷款都是用同一种方式计算，计算方式的主要区别在于等额本息和等额本金，因此可以将房贷的计算抽象为计算类。在商业贷款、公积金贷款和组合贷款的头文件中包含计算类就可以直接用计算类中的计算函数，以避免重复写相同的计算函数。房贷计算的公式通过数学方法很容易就能推导出来。

最后要实现的功能是将计算结果呈现在result\_1或result\_2界面上，由于计算结果是商业贷款类中槽函数**on\_pushButton\_clicked**()的局部变量，为了变量传递的方便，可以在商业贷款类的头文件中包含result\_1.h和result\_2.h，将result\_1和result\_2类型的变量作为商业贷款类的私有成员，直接使用result\_1和result\_2类中的公有函数，而result\_1和result\_2类中的公有函数只有一个，就是构造函数。通过构造函数将参数传递到对应ui类的LineEdit中，计算结果就显示出来了。

公积金贷款、组合贷款与商业贷款的流程大同小异，这里就不一一赘述了。

以下是程序的流程图和类图：





# 结果分析

* 1. 正确性分析

对程序结果的正确性进行测试，测试数据集包含了最大值、最小值、整数、浮点数和一些随机数，结果均与房天下的计算结果一致。

* 1. **效率分析**

经过多次测试，程序平均运行时间为0.069s,计算算法的平均运行时间为0.007s。问题的规模不断变化时，时间频度T(n)不变，时间复杂度为O(1)。

程序本身占有存储空间73.0KB，输入和输出所占用的存储空间可忽略不计，算法在运行过程中只需要占用少量的存储空间，而且不随问题规模的大小而改变，因此程序的空间复杂度也为O(1)。

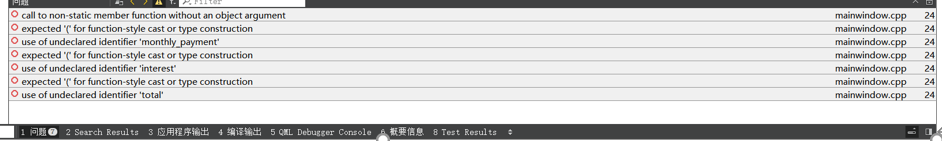
# 调试过程

一开始由于QT的使用还不熟练，调试过程中出现了不少问题。

问题1：

为了实现窗口之间的相互切换，我最初采用的方法是在商业贷款、公积金贷款和组合贷款每个类中包含其他两个类，在切换窗口时创建对应类的对象并让它显示。但这样有一个严重的问题，就是类的互相定义，而且是三个类的互相定义，用前置声明也无法解决。

解决方法：利用QT中信号和槽的机制，信号的连接、对象的创建都在main函数中进行，这样不仅解决了问题，还使代码更加简洁，同时也减少了内存的消耗。

问题2：

在商业贷款类中用到计算类中的计算函数时遇到了以上问题，主要问题是没有创建计算类的对象就使用了计算类中的公有函数。

解决方法：①将计算函数改为静态函数。②使用计算函数之前创建一个计算类的对象。

问题3：运行结果中的利息与预期不符。

解决方法：经过调试发现利率少加了百分号，将输入的利率除以100即可解决问题。

# 心得体会

一开始看到这个作业感觉最难的部分是使用QT设计人机交互界面，而且这部分我之前完全没有了解过，只能从头学起。看完QT的教学视频大概花了一个月，但真正在QT上编程时还是感觉非常生疏，这时才意识到编程不是光看视频和看书就能学会的，一定要上机操作才能融会贯通。

写完整个程序大概就花了一个星期，写这篇报告却花了一个月。写这篇报告时也学到了不少知识，比如如何用Visio画流程图，画类图，还花费了很多时间在程序正确性的证明上面。我了解到程序的正确性是可以用数学方法来证明的，采用测试方法确实可以发现程序中的错误，但却不能保证和证明程序中没有错误。不过最后我还是无法用数学语言来证明程序的正确性。

通过这次C++大作业我学到的不仅仅是C++，更是面向对象编程的思想，这在任何面向对象的语言中都是适用的。