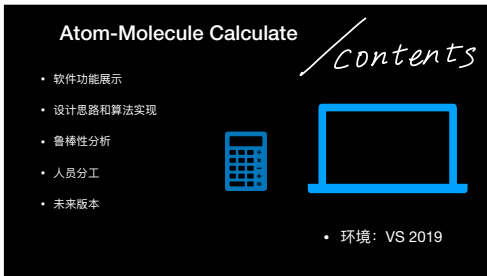
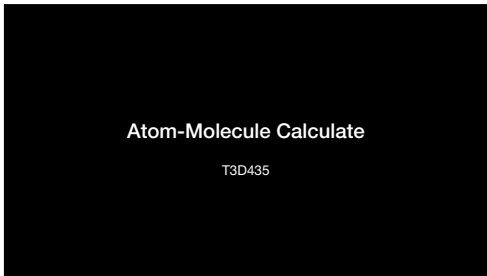
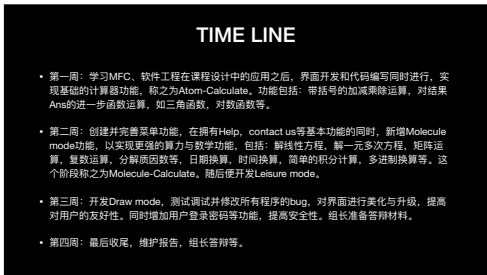


首先和大家说声抱歉，因为设备的缘故，所以这次答辩出了点小问题，因为过渡动画的不兼容，我把演示文稿转化为了视频格式，中间播放可能效果不是很好，请大家见谅。

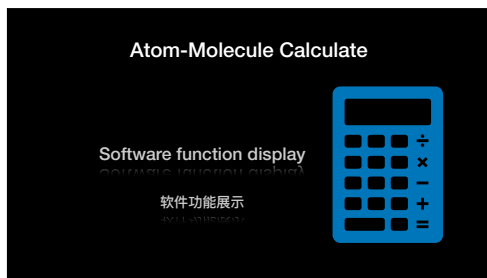
那么正式开始，我们团队做的项目是：Atom- Molecule Calculate，算是一个计算器。我们团队在开题的时候，就在想，短短四周，我们想做什么，可以做到什么程度，会不会因为时间上的仓促性导致我们团队的程序在鲁棒性上有问题，会不会在界面的美化上无能为力。在多次的商讨后，最后我们得到的答案，就是这个Atom-Molecule Calculate。



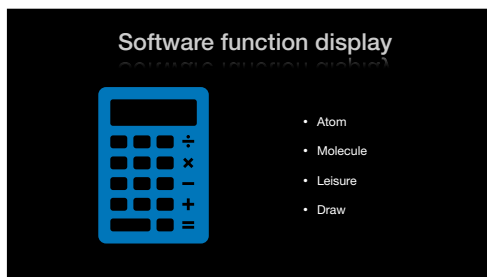
我今天的答辩分为以下几个部分：软件功能展示，设计思路和算法实现，鲁棒性分析，人员分工和未来版本。我们的软件编译环境为VS 2019，相比于VC6.0，为我们提供了更好的编译体验与对于BUG的修改指导。



这里是我们开题的时候草拟的时间线，本次项目保守估计有三周时间，经商讨，我们定下如下时间节点：



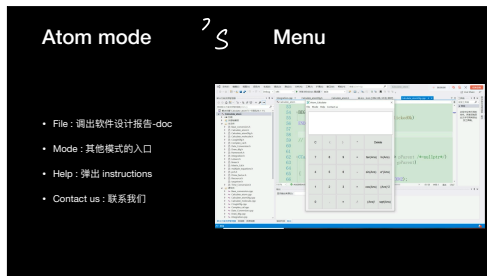
好，首先是答辩的第一个部分——软件功能展示。



我们软件的功能主要分为四大块，分别是：最基础的Atom，进阶版的Molecule，休闲模式Leisure以及画板Draw。实际上，当初开题的时候，我们并没有想那么多，刚开始只是想做出一个最基础的合格的计算器就可以了。之后我们团队在编程的过程中，发现最基础的Atom远远不够，界面开发就一个很普通的对话框，代码只要两天就能写完，算法也没有什么特别的，就是利用栈空间来运算而已。这样的结题，未免过于草率。于是，我们团队进一步开发了Molecule模式。而Leisure和Draw模式呢，则是觉得Molecule模式比较孤单，附赠给他的小伙伴。



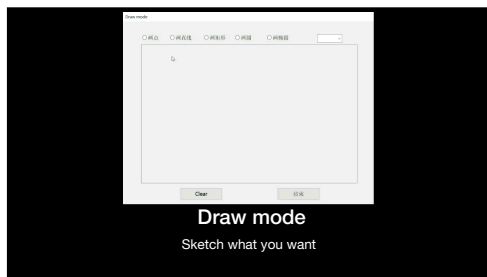
关于Atom mode，本来我们以为网上的代码能用，但是测试了很多个之后，发现有很多问题。比如说，对于小数点的不兼容，对于括号的判断有问题，抑或是，无法检测负号与减号的区分等等。于是，我们不准备采用网上的代码，或者只保留某些有用的函数并修改其有问题的BUG部分。最后，我们在Atom mode中主要实现了以下功能：支持带括号的混合运算，这里的混合运算不仅有加减乘除，还有乘方。支持对结果ANS进行进一步复杂运算，这一步没有什么特殊的算法，只是调用了C++的库函数进行进一步处理而已，比较简单。命令控件和键盘均可输入运算，这一点很显然，因为命令控件的作用很大程度上是为了显示编辑框中的文本而已。只有少数的运算命令控件才有调用函数的权利与必要。



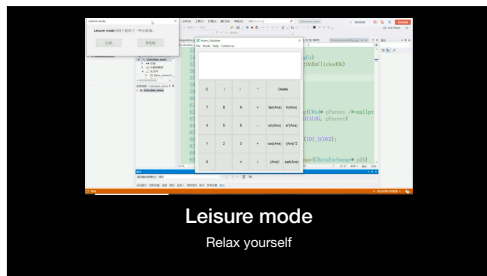
之前我们提到了其他的三个模式，基于对话框，我们设计了菜单来作为这几个模式的接口。菜单有四，File, Mode, Help, Contact us。File，很简单，调出这次软件的设计报告。Mode是其他三个模式的入口，Help则会弹出一个简略的instructions窗口出来，为程序进行简要的说明。最后，Contact us，顾名思义，就是联系我们，弹出我们编译者的联系信息。以上就是Atom mode的所有功能。



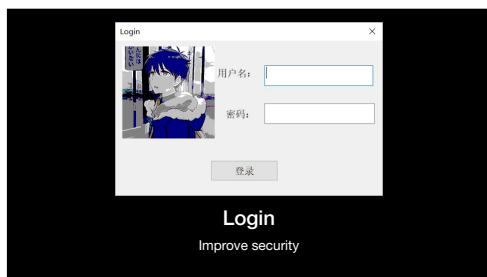
下面介绍：Molecule mode。这一部分看起来比Atom mode难一些，实际上，从算法和编译的角度上来说，Atom mode更难。这一Molecule mode对于用户输入流的限制很严格，所以功能相对于固定一些。在这个模式里面，我们有：复数计算，矩阵计算，时间换算，积分计算，解多元一次方程，质因数分解，多进制换算，日期换算，解一元多次方程以及使用说明。这里我们不再展开说，如果十个功能一个个演示过去可能时间有些不够，我们这里就只展示了图片。



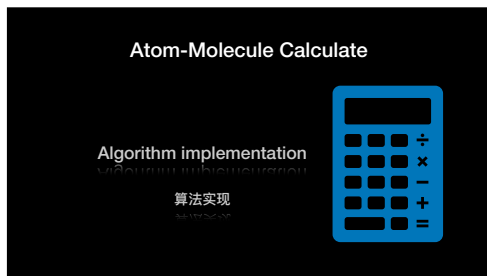
这个Draw mode是让我们很头疼的一个部分。头疼的主要原因，是因为如何建立它与主对话框的关系。实际上，这个Draw mode是独立于Atom mode和Molecule的另外一个程序，由于要用到主对话框的很多功能与性质，我们无法将它兼容到子对话框进行运行。于是，我们查阅了很多资料与解决方案，得到的结论是，将Draw mode所Realse出来的exe文件放到与Atom生成的exe文件一个路径上，利用自带的函数进行调用exe运行，便可以看起来像是其一个子对话框，假装他们是一个整体了。事实上，为了封装更多的功能，选择基于对话框的程序并不妥当。但是由于临时换成基于单文档或者多文档编程的话，学习成本上与时间上都很不划算，于是我们便利用了这个办法，将exe文件用函数进行了连接以调用，这也是为什么当Atom mode被关掉时，Molecule会自动消失，而Draw mode不会自动消失的原因。



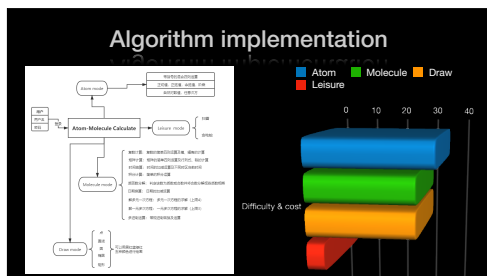
这个模式纯属是我们在第三周时改BUG改累了，想搞了玩而已。Leisure mode有两个子程序的接口，分别是扫雷和贪吃蛇。都是我们直接在网上下载的exe文件，我们直接调用了。所以说，这个mode很轻松，工作量几乎为0。但是由于是网上下载的exe文件，所以运行上有时候会有些问题。比如在某些电脑上运行时，扫雷的dll注册文件会出现问题导致无法调用。因为不是我们编写的，我们对于这一点无能为力。这个故事告诉我们，不是自己编的程序，终究还是不太听话。



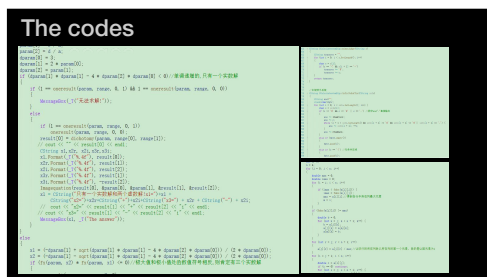
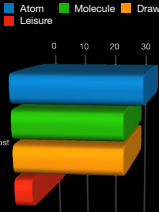
介绍完基本的功能后，来谈一谈安全性问题。为了用户的安全性，我们增加了Login对话框。只有用户名和密码与我们编译代码中的某一用户数据匹配后，才可以成功登录，拥有使用程序的权限。当然，光秃秃的登陆界面很丑吧，我们放了一个静态图片。这个图片的色彩经过我们的压缩之后可以适配mfc中的picture control了。



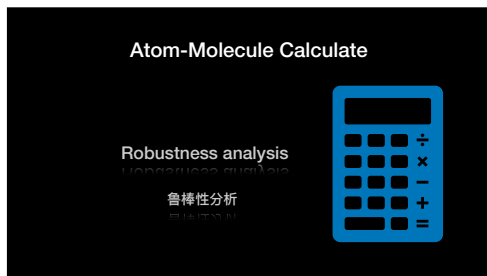
第一个部分讲的时间有点长，下面几个部分都比较短，这个部分是算法实现。



左边是我们程序设计图，右边是四个模块代码的编译难度。正如我之前说的，Leisure mode因为我们基本全是调用的网上的内容，所以无论是从编译的难度上，调试时间花费上都是最短的，占比约为不到百分之五。



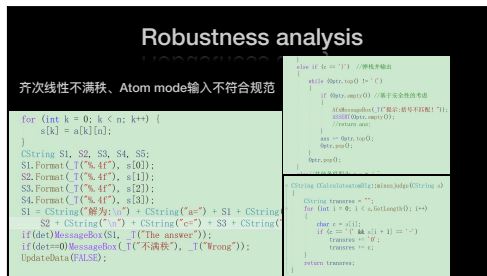
这是我们代码的截图。这里说一个题外话啊，我们平时在写代码的时候其实挺伤眼睛的，可以在网上下载一个配置颜色的，因为VS2019里面没有很护眼的背景主题。



Robustness analysis

魯棒性分析

接下来是，鲁棒性分析。在实际程序运行过程中，系统特性或参数的摄动常常是不可避免的，所以对于一个程序或者算法来说，鲁棒性分析必不可少。如果缺少了鲁棒性分析，那么程序只能说是个婴儿，永远无法长大与迭代。我们团队在第三周的主要任务就是进行不断的鲁棒性分析与修改BUG，多次地迭代与升级。

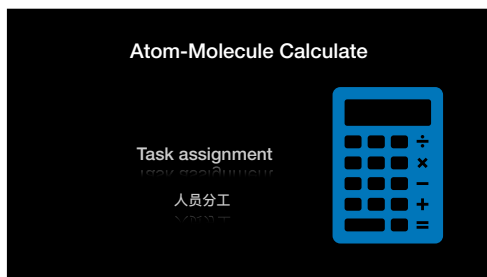


这里我就随便举几个例子来说吧：

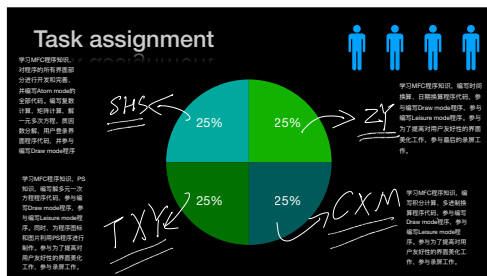
在molecule mode中，我们遇到了齐次线性不满秩的情况，也就是说是无穷多解的情况，我们利用了行列式来先进行判断是否满秩，如果是，则输出Wrong，否则则进行正常的运算。

在Atom mode里面，我们会遇到括号不匹配的情况，我们利用MessageBox来弹出小窗，提示用户的错误。

判断减号和负号的时候，我们新增了函数minusjudge，对其进行了具体情况的甄别。



这一部分是人员分工。



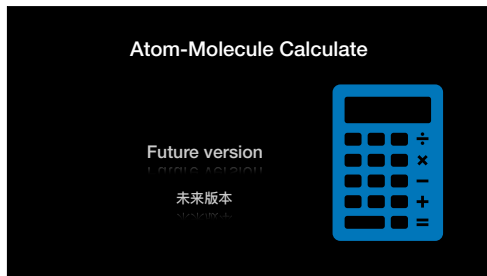
在本次任务中，

我负责的主要是：学习MFC程序知识，对程序的所有界面部分进行开发和完善，并编写Atom mode的全部代码。编写复数计算，矩阵计算，解一元多次方程，质因数分解，用户登录界面程序代码，并参与编写Draw mode程序。

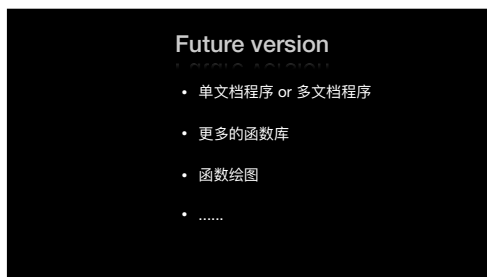
张扬：学习MFC程序知识，编写时间换算、日期换算程序代码，参与编写Draw mode程序，参与编写Leisure mode程序。参与为了提高对用户友好性的界面美化工作，参与最后的录屏工作。

陶星宇：学习MFC程序知识，PS知识，编写解多元一次方程程序代码，参与编写Draw mode程序，参与编写Leisure mode程序。同时，为程序图标和图片利用PS程序进行制作。参与为了提高对用户友好性的界面美化工作，参与录屏工作。

车旭明：学习MFC程序知识，编写积分计算、多进制换算程序代码，参与编写Draw mode程序，参与编写Leisure mode程序。参与为了提高对用户友好性的界面美化工作，参与录屏工作。



未来版本，是我们团队对于Atom- Molecule Calculate寄予的未来期望。当然，我们不会梦想它成为什么Matlab就是了。

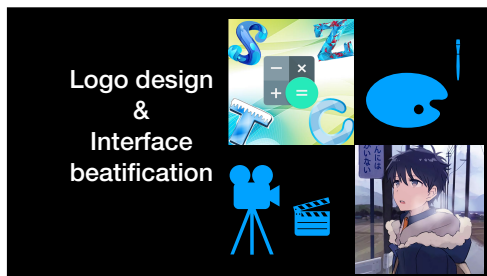


未来的版本中，我们希望能实现向单文档程序或者多文档程序转型，突破对话框的限制。

下一步，我们可以拥有一个函数库，便于实现求导等功能。

有了函数库，我们就可以配合Draw mode，创建一个笛卡尔坐标系，进行函数绘图。

等等。



值得一提的是，我们本次mfc项目中，界面的美化和独占性没有丝毫的放松。举个例子，我们的优秀队员设计出了属于我们的软件标志，不再是默认的mfc三个立方体了。为了更改图标，我们需要先在ps软件中绘制出自己要的图案，然后在网站的帮助下，将其转换为ico的格式，之后用icon导入程序，将程序本身的framework给替代掉，最后利用一个bat刷新图标功能清空了系统的缓存图标，从而最后得到了我们想要的独有图案。在程序中插画也是一样，我们需要先用ps将文件转换为bitmap并限制大小，之后压缩色彩，将其劣质色彩版本导入到bitmap文件当中，利用picture control控件进行充分利用。

THANK YOU.

谢谢。
