### 运行环境

Win 10/11

Qt 6.5.2 MinGw 64-bit

Qt 6.5.2 MSVC2019 64bit

### 编程说明

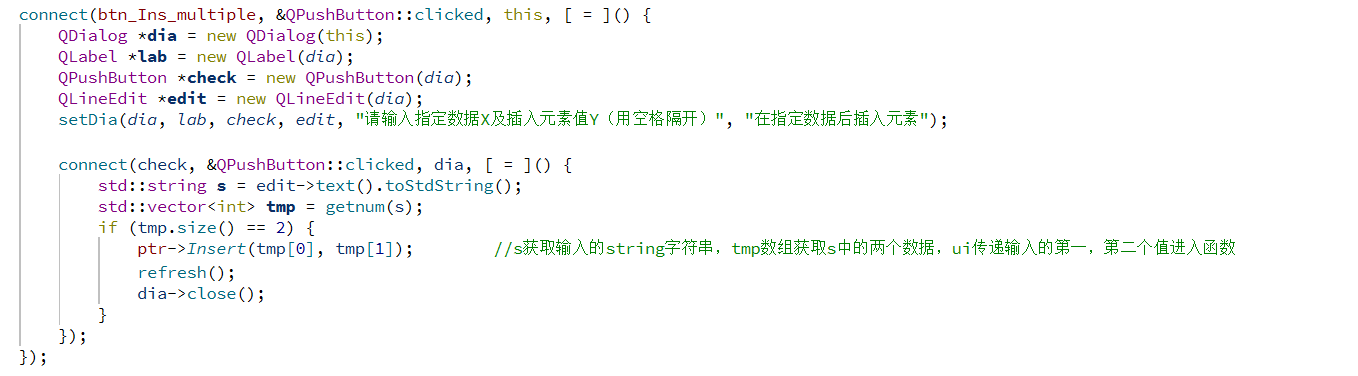
请按照模板所给定义实现单链表和顺序表，将相应函数置于Seqlist.h 和linklist.h中。

模板中， 类list (list.h)为Seqlist (Seqlist.h)和Linklist (linklist.h)类的基类。三种类的具体定义见代码。图 中为Linklist的声明。



代码中，注释所指出的部分请按照所给定义来实现，否则程序可能无法正常运行。

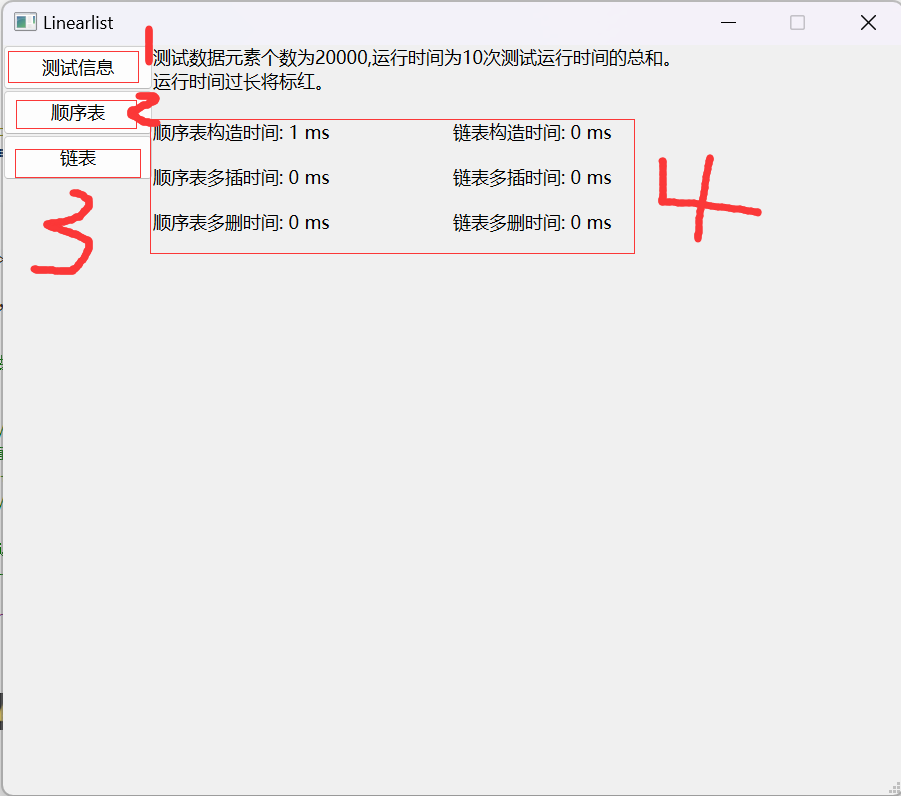
注意， c++中模板类(使用了template的类)声明与实现必须是同源的。若函数实现直接置于cpp文件 中，将出现编译错误。该问题可以通过 export 关键字或模板显式实例化解决。因为Qt不支持export关 键字，故在该代码中使用了模板显式实例化。具体请见Seqlist.cpp和linklist.cpp。



这是ui界面与函数界面的连接函数，基本原理为s获取输入的string字符串，tmp数组截取数据，将tmp的数据作为参数传入函数。

### 关于UI的说明

主窗体**(mainwindow.h)**

主窗口为类MainWindow，对应文件mainwindow.h 。如图所示，主窗口包含共包含四个控件。

控件1~3均为QPushButton类，分别对应对象 btn\_info , btn\_seq , btn\_link ，用于实现对控件4窗 体切换的控制。

控件4为QStackedWidget ，对应对象 CenteralWidget ，该控件中包含三个窗体 wid\_info , wid\_seq , wid\_link 。

测试信息窗体**(infowidget.h)**

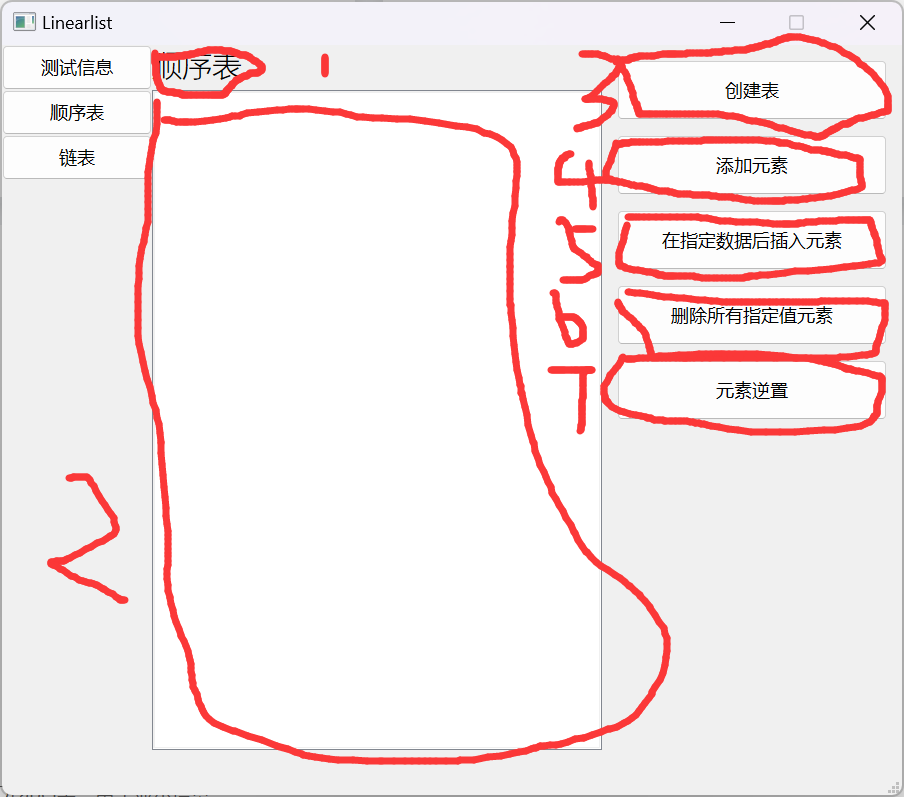
测试信息窗体为类InfoWidget，也既上图控件4中显示的窗体。包含数个QLabel控件，用于显示对应信 息。

在该类中，将对顺序表和链表进行复杂度测试，运行时间过长的部分将用红色标出，顺序表和链表的各 功能算法时间复杂度都不应超过  。

上图中，循序表多插为 。

顺序表窗体**(seqwidget.h)**

对应的类为SeqWidget，共包含7个控件。



控件1为QLabel类，用于显示信息。

控件2为QListWidgett类，对应对象widget用于显示顺序表中对应的元素。

控件3~7为QPushButton类，对应顺序表的各个函数功能，分别对应对象：

btn\_creat , btn\_push, btn\_Ins\_multiple , , btn\_Del\_multiple , , btn\_reverse

上述按键控件在点击后将调用顺序表对应的函数，并调用函数 refresh()刷新控件2widget 中的内

容。若有需要输入的数据，则弹出QDialog对话框，显示相应信息要求用户键入数据。

链表窗体**(linkwidget.h)**

与顺序表窗体类似，不再赘述。

若想增加功能及按钮，可以参考如下：

UI的修改  
  
本模板中没有使用QT中的.UI文件，所以UI文件中是没有内容的， 所有窗体的UI均通过代码实现，UI的修改也需要通过修改代码来实现。

以Seqwidget为例，在Seqwidget.h中, 定义了在该窗体中所用到的各个控件类的指针:在Seqwidget.cpp的Seqwidget（)函数中， “new'出了各控件的对象，同时通过resizeO和move()函数对各控件的大小和位置进行修改。connect 函数则是将各个控件上的事件(如点击)和发生该事件后的行动关联起来，该函数的最后-一个参数是一一个函数指针,即发生事件后响应的函数，在改模板中大多使用了lambda表达式实现该函数。

将光标移动到某-控件或QT实现的函数上， 按下F1即可调出该控件或函数的说明文档，可以根据说明文档学习如何使用该函数或控件。除了使用说明文档外，可以网络来学习某一控件的用法。在学会该控件的用法后,修改相应的代码即可。

**Hint**

为保证程序的正常运行，请严格按照Seqlist.h和Linklist.h中所给的声明实现顺序表和链表！！