车辆管理问题 作业

（采用Visual Studio 2017 15.6.4 Windows SDK 10.

这是这学期的第二次上机作业。因为之前在做别的，又提前在做下一次的作业花了不少时间，这次的作业没有花费过多的时间，很多地方都只是冲着“实现”去的，而没有在意效率。因为做得比较急，注释也比较少，没有及时纳入git版本控制。

感觉主要的收获是有了一定复杂的与人的命令行交互的处理经验，以后可以更好地做功能性的应用，对写cmd的shell也有一定帮助，虽然没有采用之前写过的底层处理命令的操作（主要是对非ASCII码的支持和对非标准输入的兼容，对路径里面双引号和斜杠转义的支持一类）。

总的来说还是合理地完成了任务，但没有特别出彩的地方，不能说特别满意。

0、程序要完成的内容

完成简单的最多100辆车的信息和不确定条目的违章记录的管理。主要是其中每一种内容的查找和编辑。最后还要文件输入和保存信息以备下次使用。

1、最终程序的结构是这样的：

用一个命名空间封装整个过程，所有操作封装在一个操作类里，由一个和人交互的命令处理函数作为整体调用的接口，其它内容都是private的，禁止外部调用，避免非法操作。用命令来调用对内存中信息的处理完成任务。

数据结构上直接采用了顺序表+链表的结构，用顺序表存储车辆信息，链表存储某一辆车的违章信息。违章信息条为操作简便，使用栈的数据结构。

在操作上，把查找违章和车辆在顺序表中的位置写成了底层的函数，不同的操作在查找时都调用这个函数，查找后进行操作或者输出。

为了让程序可以保存和退出，写了两个命令，格式化输出内存中保存的各个条目的信息到一个固定文件。在构造函数中调用加载保存内容的函数，从而保证程序可以打开时加载之前保存的信息。在需要时可以保存内存中的信息到文件，或者加载文件中保存的内容。

为了方便人输入大量的违章信息和车辆条目而不需要在命令行一次性输入正确，写了两个解释文件中命令的函数，可以按文件中逐行添加条目信息。毕竟c++ 的输入文件流和cin是用法相同的（以后可以考虑用类似的方法，给我的cmd shell 添加解释执行脚本的功能）

当然还有删除的功能，在顺序表中删除一辆车和它相关的违章信息。以及将一个违章条目标记为已处理的功能。一般已处理不会再变为未处理，就没有管这个。

2、一些遇到的问题和解决方法

车辆信息的顺序表其实可以在查找信息上使用折半查找提高效率，但这需要按顺序存储车辆信息，并且每一个搜索的函数都要大大增加代码量，得不偿失。

还有一种方法是用车辆的下标当作编号，但是这肯定和实际中车辆牌照不可能是0-100一样，只能算理想情况，只能用作底层代码，并且如果用这样一个下标作识别码，人输入车牌号时还是要进行查找，而不能随机存取，故没有采用。

为动态分配的空间可以释放，每一级有析构函数 “delete” 下一个节点。这种做法不少唯一的，比如也可以把头节点删掉，然后让下一个作头节点，然后重复操作。但是这样的操作不能适应一般情形。

如果数据结构是树一类后继不唯一的情况（这是在写这个程序之前在写书目词索引表时为了加速查找，使用字典树时想到的），要进行如此的删除就也必须要用到栈的数据结构。而最方便的方法就是递归调用一个函数，而析构函数里调用会自动调用析构函数的delete正好可以完成这个任务，就不用自己写一个函数完成了。现在的操作系统，程序一般能有1mb左右的栈，只要不进入死循环，程序的栈一般还是够函数来调用的。

在比较重要的查找方式上采用的是比较简单的按顺序查找。但是因为每一个违章不一定只有一辆车参与，而每辆车是否处理有不一定相同，所以要查找的内容和直接的结果并不直接对应。

为此暂时创建了一个违章条目表，然后所有满足条件的违章记录一一查看违章编号并把不同的满足条件的违章编号和一个编号联系起来，然后把所有对应的车辆找到，从而建立违章条目和相关车辆的关系。

3、一些可以改进的地方

在查找中使用的是顺序查找，车辆条目没有顺序。可以考虑改为按一定顺序排布条目，然后折半查找车辆信息。在车多于7个时开始有实际意义。

查找的返回值目前是c++ 的vector（元素是自定义的查找到的条目结构体），而vector并不一定是一个足够好的容器，可以考虑修改。

目前文件输入对长的路径和多文件输入支持不好，文件路径默认为“add violation”或者“add car”之后去掉马上连上的所有空格后的整个一行（getline(cin, string)）。这虽然可以支持带空格的文件路径，但不支持带引号的路径（和cmd中行为不同）或者多个文件进行输入。这个需要专门写一个处理文件路径的函数解决。

对一些常见的非规范输入不够兼容，对一些失败的输入，失败原因没有进行细分，并且输入格式比较繁琐，且没有提示。这需要一个manual的提示命令解决（man或者 –help命令输出提示）。

4、总结

这次上机作业实在是有点赶了，主要是这段时间出了点小意外，并且之前在下一个作业上提前多花了点时间。

感觉赶着做，主要问题就是程序的结构不好控制，最好还是提前想好整个结构，然后一点点来，不然最后改结构是很麻烦的。但是的确，因为水平有限，并不可能提前都知道，很多时候还是要一边查资料一般码代码，所以重点还是要提高自身能力。