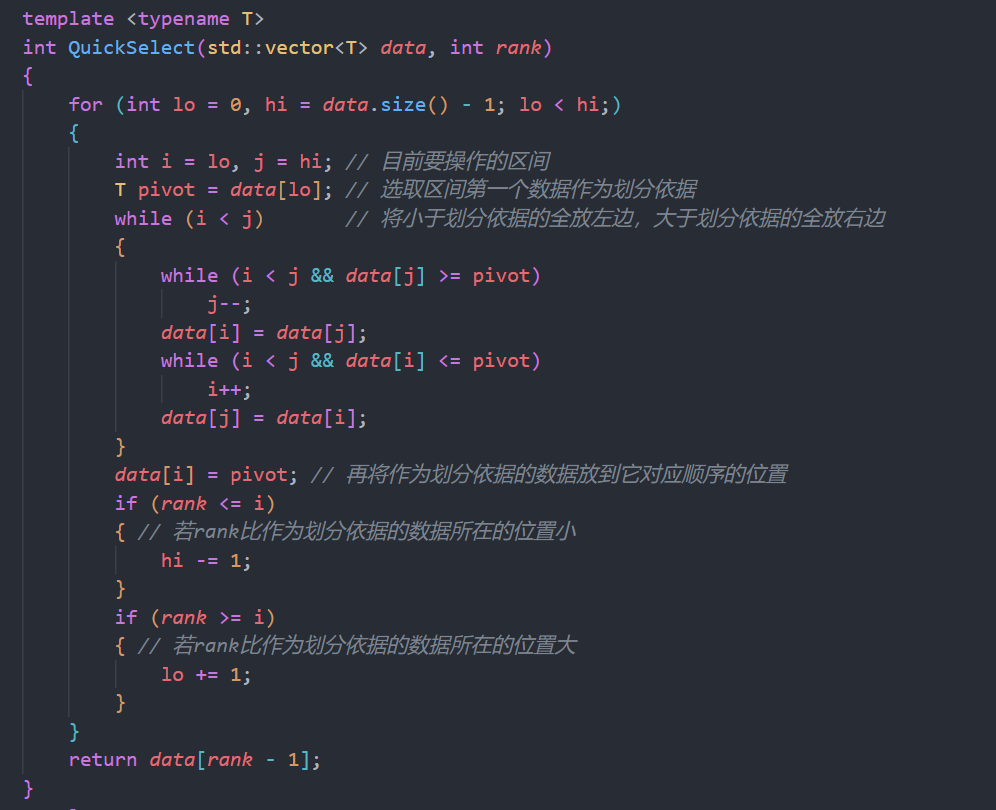
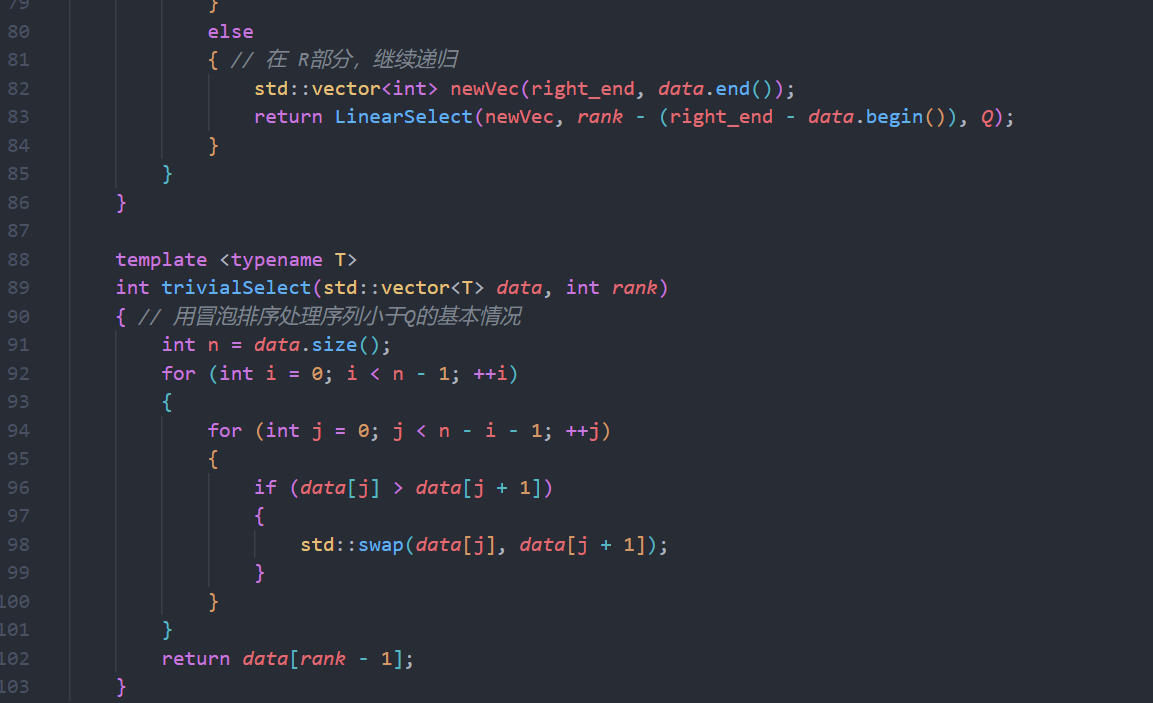
1. 具体实现与老师ppt几乎一致，下图中注释详细解释了两种算法的实现逻辑

QuickSelect:



LinearSelect:

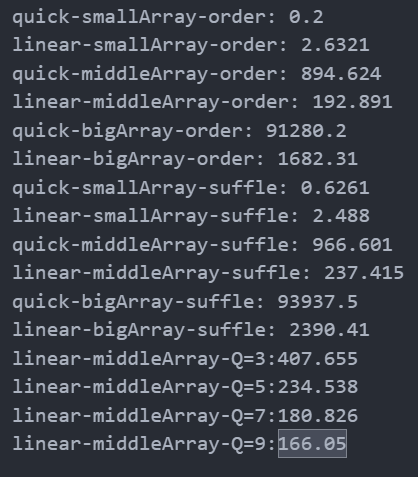


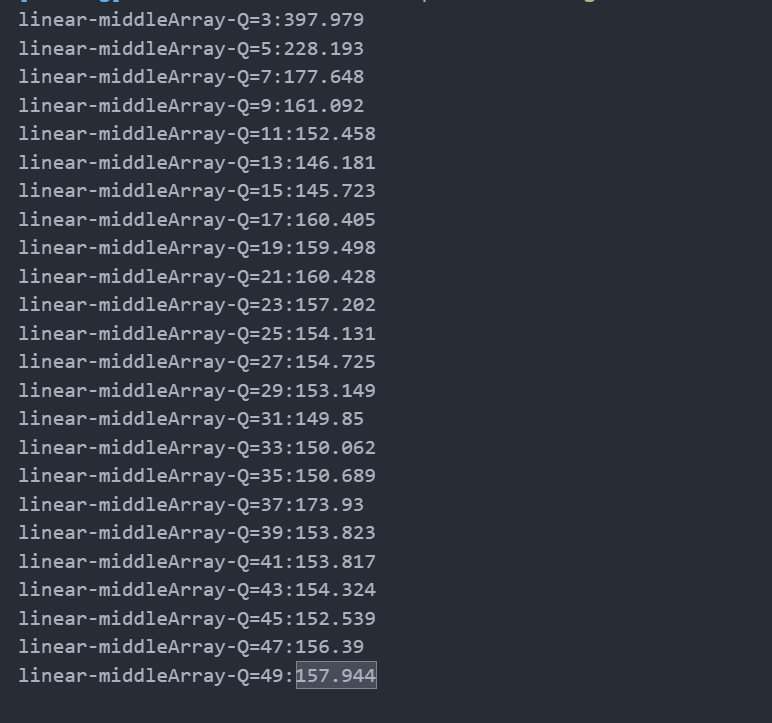
2.1针对数据规模和乱序性设计了12组实验，每组实验进行10000次，取平均执行时间。针对数据规模，设计了10,1000,10000三个不同规模，针对乱序性，设计了顺序和乱序（shuffle）两种情况。实验代码如下，先建立了三个不同大小的vecotr集合，并顺序插入数据，然后对每种数据规模的数据分别使用QuickSelect和LinearSelect并记录，进行6次实验；然后将三个集合都乱序掉，再次重复6次实验。

2.2针对LinearSelect的Q的大小，Q取值从3开始，到50结束，每隔两个数字取一个字（3、5、7、9……49），对于每个Q，进行10000次实验，取平均值

1. 测试结果：

代码运行结果图：





实验结果分析：

1. 无论在顺序还是乱序情况下，当数据规模非常小时，QuickSelect比LinearSelect性能好，但在数据规模较大时，LinearSelect比QuickSelect性能要好。
2. QuickSelect在任何数据规模下，乱序下算法的性能都要稍差于顺序下的性能；但LinearSelect算法在数据规模非常小时，乱序下算法要稍优于顺序下的性能，其余情况乱序下算法要稍差于顺序下的性能。