# 数据结构项目十文档

软件学院 14级1班

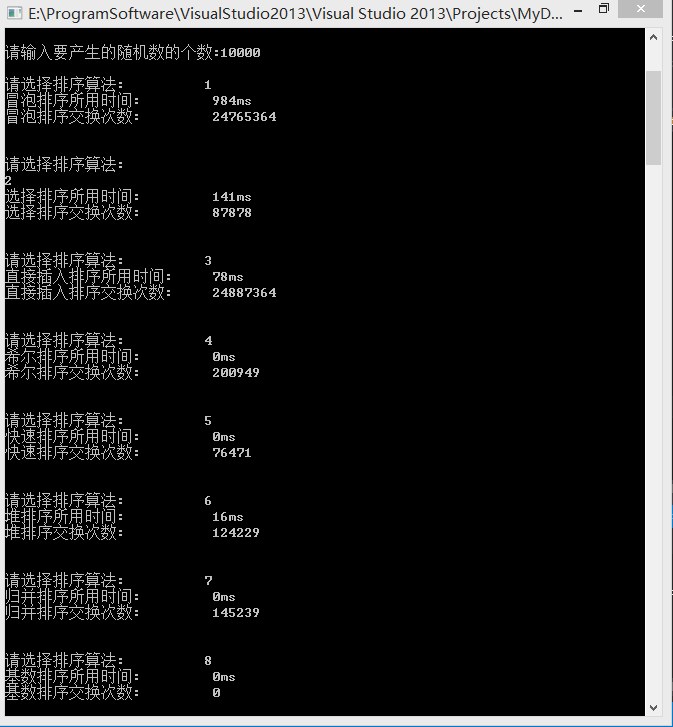
吕金华

# 1.项目背景：

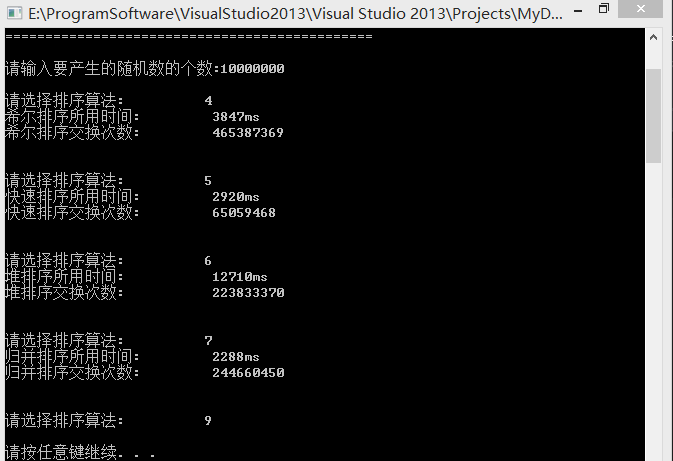
随机函数产生10000个随机数，用快速排序，直接插入排序，冒泡排序，选择排序的排序方法排序，并统计每种排序所花费的排序时间和交换次数。其中，随机数的个数由用户定义，系统产生随机数，并且显示他们的比较次数，排序算法包括冒泡排序，选择排序，直接插入排序，希尔排序，快速排序，堆排序，归并排序和基数排序。

# 2.操作示例：

数据量为1w



数据量为1000w



# 3.项目特色：

1.将所有算法集成到一个SortMachine类中，方便使用

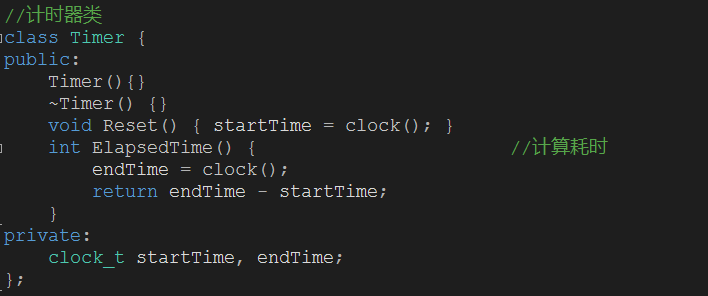
2.编写了计时器类以及测试类

3.使用模板参数，使得算法独立于变量类型，方便代码复用

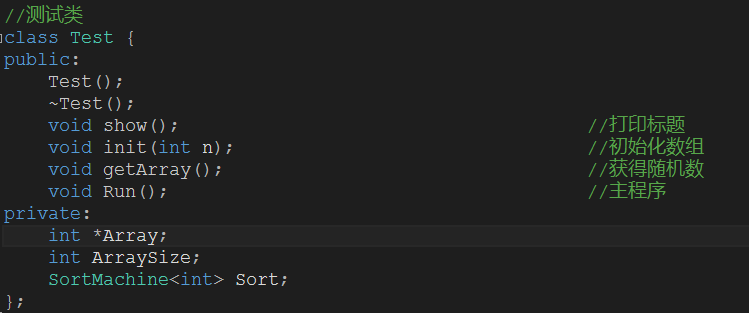
4.优化了快速排序和归并排序，使它们的速度在一般情况下比qsort和sort快

# 4.代码分析：

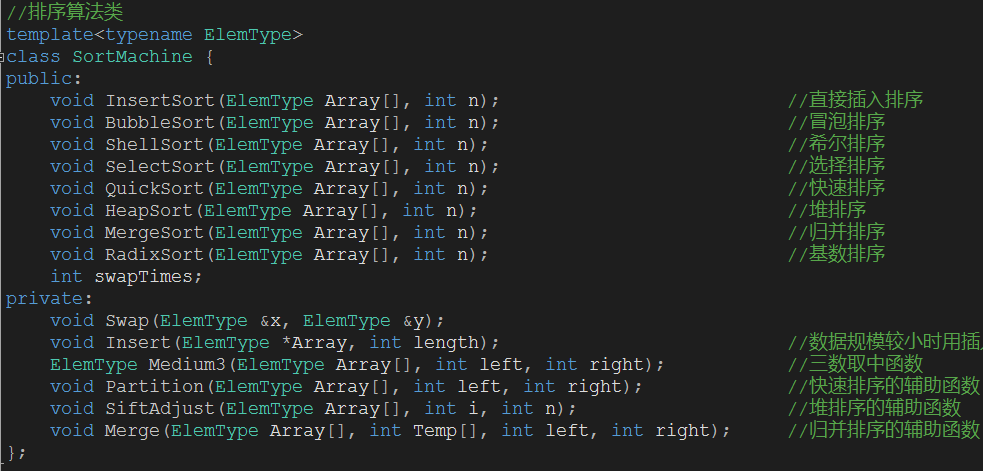
## 1.Timer类



## 2.Test类



## 3.SortMachine类



# 5.算法分析：

1. 快速排序选取不同的优化方式效率相差真的可以很大。一开始直接用Array[0]作为主元，最差情况下跟冒泡差不多快。后来用了优化，用rand()函数随机选择主元，但是rand()自身比较耗时。于是我使用了三者取中法，并且在递归到子区间长度小于一个值时，调用插入排序，得到了比较好的效果。在100w数据规模下，比STL的sort要快上十倍，比qsort要快上两三倍。这个值经过我的测试，取30的时候比较好。
2. 归并排序在一般情况下，表现的和快排一样快，而且很稳定。用插入优化之后性能优于快排。
3. 堆排序比快排差不多慢一倍。
4. 希尔排序作为插入排序的优化算法，表现得比快排还快，我将数据量加到10000w发现希尔排序还是比快排快。
5. 基数排序理论上时间复杂度为O(d\*(n+r))，在这个题目中d较小，差不多为线性的，速度也很快。