**实验总结：**

课堂学的和动手实践是两个难度，将理论应用到实践的过程中会有很多障碍。以下是遇到的困难与解决方案：

1. 如何生成相同组成，相同长度，但是ADD不同的序列：

先随机生成一个序列，计算ADD值，将此值记为最大、最小ADD的初值，然后对此序列进行变换（用random\_shuffle增加ADD,或partition减小ADD），每次变换后重新计算ADD，直到新的ADD值大于最大ADD一定值或小于最小ADD一定值，保留此序列进行排序测试，重新计最大ADD或最小ADD，并以此序列为基础继续生成后续序列。

1. 如何获取次近点对：

把例子中每次递归查找左、右、中三个区域的最近点对改为获取左、中、右三个区域的最近和次近点对，在这六个点对中比较得出此区域的最近和次近点对，层层返回，最后得到整个区域的最近和次近点对

1. 如何处理中位线穿过多个点的情况：

课本默认中位线不穿过任何一个点，因此选择了直接在排序后取中间两点平均值的形式。但实际中可能有2个甚至多个点的横坐标相同且恰好等于中位数，此时若仍取两点平均值，中位线就会穿过这些点，做划分的时候会无法分辨此点应划分到P1还是P2。

为此我设计了两种解决方案

1. 若中间两点的横坐标相等，则几下他们纵坐标的平均值，若比较的时候遇到某点被中位线穿过，则比较它的纵坐标与该平均值的大小关系，若小于平均值，则置于P1，反之至于P2。
2. 设置计数位eq,初始化为0，意义为在中位线的P1侧应划分多少被中位线穿过的点。若中间两点横坐标相等，则置eq为1，同时从x[m-1]开始扫描至区间结束或遇到不被中位线穿过的点，此间每扫描到一个被中位线穿过的点，eq自加。分割时将eq个被中位线穿过的点划分到P1，其余划分到P2。由于X集合有序，方案有效。

经过权衡我选择了方案2.