Problem 1

(a) Insertion Sort

[314, 512, 004, 999, 023, 042, 613, 109, 001, 123, 666]
[004, 314, 512, 999, 023, 042, 613, 109, 001, 123, 666]
[004, 023, 314, 512, 999, 042, 613, 109, 001, 123, 666]
[004, 023, 042, 314, 512, 999, 613, 109, 001, 123, 666]
[004, 023, 042, 314, 512, 613, 999, 109, 001, 123, 666]
[004, 023, 042, 109, 314, 512, 613, 999, 001, 123, 666]
[001, 004, 023, 042, 109, 314, 512, 613, 999, 123, 666]
[001, 004, 023, 042, 109, 123, 314, 512, 613, 999, 666]
[001, 004, 023, 042, 109, 123, 314, 512, 613, 666, 999]

(b) Quik Sort

(Pivot shown with underline)
[004, 023, 042, 109, 001, 123,/ $\underline{314}$,/ 512, 999, 613, 666]
[004, 023, 042, 001, /$\underline{109}$,/ 123,/ $\underline{314}$,/ 512, /$\underline{613}$,/ 666, 999]
[004, 001, 023, /$\underline{042}$, /$\underline{109}$,/ 123,/ $\underline{314}$,/ 512, /$\underline{613}$,/ 666, /$\underline{999}$]
[001, /$\underline{004}$,/ 023, /$\underline{042}$, /$\underline{109}$,/ 123,/ $\underline{314}$,/ 512, /$\underline{613}$,/ 666, /$\underline{999}$]

(c) Radix Sort

0
1 001
2 512 042
3 023 613 123
4 314 004
5
6 666
7
8
9 999 109

0 001 004 109
1 512 613 314
2 023 123
3
4 042
5
6 666

```
7
8
9  999

0  001  004  023  042
1  109  123
2
3  314
4
5  512
6  613  666
7
8
9  999
```

Problem 2

(a) Add extra information of the original position of the object in the array. And when the comparater do a comparison, it compares the object's original key first and go on to compare the extra information if it's equal.
(b) Example: Quick sort 7 1 3 4 1
Choose 3 as pivot, and the first step is swap 1 and 7, thus changing the relative position of the two "1".

Problem 3

O(logN)

Problem 7

Use two heaps, one min heap and one max heap. And when encounter a new numberin the stream, put it in the min heap first. And remove the minimum from the min heap and put the minimum of the min heap into the max heap. The medium should be the maximum in the max heap.