Question 1: Problem 3

Stable: Bubble sort, Insertion sort, Merge sort, Counting sort, Radix Sort
Not Stable: Quick sort, Heap sort, Selection sort
Can be made stable: All non-stable can be made into stable by taking into account the positions of the elements prior to the sort

Question 2: Problem 4

Yes, $SELECTION$ with groups of 3 does find the k-th smallest element, however it will not run in linear time since we don't reduce the subproblems as efficiently as with groups of 5. With groups of 3 we are still left with subproblems of size n whereas with groups of 5 and larger the problem get reduced to size of less than n.

Question 3: Problem 7

QUICKSORT: Depending on implementation, chooses either the first or last element of array as pivot.
SELECTION: Just as QUICKSORT, chooses either the first or last element as pivot.
RANDOMIZED SELECTION: The pivot is chosen at random uniformly using a random number generator from (start,end)

Question 4: Problem 21

For a vertex to be a cut vertex as a function of its discovery time, for two nodes, u, v, where u is a parent and v is a child. Let $d[v]$ be the discovery time for a node v. Let $L[u]$ be the lowest node that can be reached from any vertex. Then if $L[v] \geq d[u]$ then u is a cut vertex.