Exam 2 Review

The SELECTION problem:

- $SELECT(A, k)$: returns the ith smallest element in A (where $k = 1...n$)
- QUICK-SELECT: runs in $O(n)$ time expected, $O(n^2)$ worst-case
- SMART-SELECT: runs in $O(n)$ time worst-case
- Note: works with duplicate keys; in practice QUICK-SELECT preferred

Techniques: Examples in lectures/labs/homeworks:

Divide-and-conquer:

- Mergesort, Karatsuba large-integer multiplication, Strassen matrix multiplicatin
- Maximum subarray, finding majority element, counting inversions, finding quantiles
- Also binary-search-like algorithms (finding singleton element, finding peak, etc)

Dynamic programming:

- Fibonacci, Playing a board game, Rod cutting, Knapsack
- Robbing a house, unbounded knapsack, pharmacist
- Subset sum, string shuffling
- Optional: Playing a game, skis and skiers, longest common subsequence (LCS), longest increasing subsequence (LIS)

Greedy:

- Greedy counter-examples
- Activity selection
- Fractional knapsack; Pharmacist when all bottles have same cost
- Art gallery guarding
- Optional: Turkeys, skis and skiers (when $n = m$), matching points on a line

For a given technique, the principles are the same, but it is instantiated differently and creatively in each specific problem. Go through this list and reflect on each problem; use the code provided, or write your own, to check your understanding.