It is recommended that you read through the exam before you begin. Answer all questions in the space provided.

Name: ______________________________

Answer whether the following statements are true or false and support your answer with a proof sketch or counter-example in the space provided.

1. [TRUE / FALSE] There is no algorithm that solves the *Convex Hull* problem in $o(n \log n)$. [5 pts]

2. [TRUE / FALSE] If the pivot is randomly chosen the worst case complexity of the QuickSelect algorithm shown in class is $O(n)$. [5 pts]

3. [TRUE / FALSE] Every binary tree with n leaves has height $O(\log n)$. [5 pts]

4. [TRUE / FALSE] When using dynamic programming tabulation will always result in a faster algorithm than memoization since it avoids the overhead of recursive calls. [5 pts]

5. What are the two key features that a problem needs to exhibit that makes it a good candidate for dynamic programming? [10 pts]

6. Explain what would happen if a dynamic programming algorithm is designed to solve a problem that does not have overlapping sub-problems. [10 pts]

7. Prove that finding the sum of an array of n integers requires $\Omega(n)$ time. [15 pts]

8. Suppose we want to make change for n cents, using the fewest number of coins of denominations 1, 10, and 25 cents. Consider the following greedy strategy: suppose the amount left to change is m; take the largest coin that is no more than m; subtract that coin's value from m and repeat. [10 pts]

   Prove or disprove that this algorithm always outputs an optimal solution.

9. Consider the *Fractional Knapsack Problem*: 'Given n items with values and weights and a knapsack with capacity C. You may take fractions of each item and as many of each item as you want. Find the selection of items that maximizes the possible value that can be fit in your knapsack.' Describe an efficient greedy strategy for this problem and prove that your greedy choice strategy works. [15 pts]

10. What is the worst-case complexity of the following algorithm? Explain your answer. [10 pts]

```
GRAHAMSCAN($Q$)
let $p_0$ be the point in $Q$ with the lowest $y$-coordinate
let $\langle p_1, p_2, ..., p_m \rangle$ be the remaining points in $Q$
sorted by polar angle around $p_0$
if $m < 2$
    return "convex hull is empty"
else let $S$ be an empty stack
    PUSH($p_0, S$)
    PUSH($p_1, S$)
    PUSH($p_2, S$)
    for $i = 3$ to $m$
        while the angle formed by $p_i$ and the top 2 elements
        of $S$ makes a non-left turn
            POP($S$)
        PUSH($p_i, S$)
return S
```

11. In class we described an algorithm to solve the 2D Closest Pair problem in $O(n \log n)$ time. Briefly describe a way to extend this solution to solve the 3D Closest Pair problem that still runs in $O(n \log n)$ time. [10 pts]