Directions: For each of the first four problems, a "high level" greedy strategy is given. For some of the problems, the strategies give a correct (optimal) solution, and for others, it sometimes gives incorrect (suboptimal) solutions. For each, decide whether the greedy strategy produces optimal solutions. If it is, give a proof that it is correct, then describe what data structures and preprocessing you would use to give an efficient version, and give a time analysis. (10 points, correctness proof, 10 points efficiency and correct time analysis)

If it is not correct, give a counter-example showing the strategy is incorrect. Then still give an efficient version, as a heuristic. (10 points, counter-example, 10 points efficiency and correct time analysis)

Caravan stops You are organizing a caravan crossing a desert. Your path is fixed. The caravan can only travel $m$ miles between stops at oases. You are given a list of oases $Oasis[1..n]$, each with its distance $d_i$ in miles from the starting point. The last oasis, $O_n$, is the destination. You wish to choose the minimum size set of stops subject to the constraint that there are no more than $m$ miles between consecutive stops.

Candidate greedy strategy: Treat the start as an oasis with $d_i = 0$. At each stop, at oasis $Oasis[i]$, if the destination $d_n \leq d_i + m$, go there directly. Otherwise, find the oasis $j$ of maximimum $d_j$ subject to $d_j \leq d_i + m$. Make $j$ your next stop, and repeat.

Maximum independent set An independent set in an undirected graph $G = (V, E)$ is a set of nodes $S \subseteq V$, so that no two nodes in $S$ are adjacent in $E$. i.e., if $\{x, y\} \in E$, we cannot have both $x$ and $y$ in $S$. The maximum independent set problem is to find a largest independent set in a given graph.

Candidate greedy strategy: Pick the node $x$ of smallest degree, and put $x \in S$. Remove $x$ and all of its neighbors from $G$. Repeat until no nodes are left.

Oxen pairing Consider the following problem: We have $n$ oxen, $Ox_1, \ldots, Ox_n$, each with a strength rating $S_i$. We need to pair the oxen up into teams to pull a plow; if $Ox_i$ and $Ox_j$ are in a team, we must have $S_i + S_j \geq P$, where $P$ is the weight of a plow. Each ox can only be in at most one team. Each team has exactly two oxen. We want to maximize the number of teams.

Candidate Greedy Strategy: Take the strongest and weakest oxen. If together they meet the strength requirement, make them a team. Recursively find the most teams among the remaining oxen.
Otherwise, delete the weakest ox. Recursively find the most teams among the remaining oxen.

**Cookie assignment** Consider the following problem:

You are baby-sitting $n$ children and have $m > n$ cookies to divide between them. You must give each child exactly one cookie (of course, you cannot give the same cookie to two different children). Each child has a greed factor $g_i, 1 \leq i \leq n$ which is the minimum size of a cookie that the child will be content with; and each cookie has a size $s_j, 1 \leq j \leq m$. Your goal is to maximize the number of content children, i.e., children $i$ assigned a cookie $j$ with $g_i \leq s_j$.

Candidate Strategy: Look at the greediest child. If the largest cookie makes the child content, give the child the largest cookie. Otherwise, give the child the smallest cookie.

**Implementation** Implement the greedy algorithm for maximum independent set and test it on random graphs where each possible edge is in the graph with probability 1/2. What is the average size of the independent set it finds for graphs of different sizes? (Try $n$ as many powers of 2 as you can.) How do you conjecture the size will grow as a function of $n$?