1.	**Activity Selection Problem** Problem Description: Suppose we have a set $S = \{a_1, a_2, \ldots, a_n\}$ of n proposed activities that wish to use a resource, such as a lecture hall, which can serve only one activity at a time. Each activity a_i has a start time s_i and a finish time f_i, where $0 \le s_i < f_i < \infty$. If selected, activity ai takes place during the half-open time interval $[s_i, f_i)$. Activities a_i and a_j are compatible if the intervals $[s_i, f_i)$ and $[s_j, f_j)$ do not overlap. That is, a_i and a_j are compatible if $s_i \ge f_j$ or $s_j \le f_i$. In the activity-selection problem, we wish to select a maximum-size subset of mutually compatible activities.	10
2.	**Finding Minimum Stops [Link1]** Suppose you were to drive from A to B, which is **D** miles away, along a straight road. Your gas tank, when full, holds enough gas to travel **m** miles, and you have a map that gives distances between gas stations along the route. Let **d1** < **d2** < … < **dn** be the locations of all the gas stations along the route where **di** is the distance from St. Louis to the gas station. You can assume that the distance between neighboring gas stations is at most **m** miles. Your goal is to make as few gas stops as possible along the way. Give the most efficient algorithm you can to determine at which gas stations you should stop. Write a code to solve this problem using a **greedy algorithm**. Keep the time complexity of your code O(n).	10

Sample input **D** **m** **n** **d1 d2 … dn**	Sample output
20 10 8 2 4 5 8 12 14 16 19	stop at gas station 4 (8 miles) stop at gas station 7 (16 miles)
20 10 4 2 8 12 14	stop at gas station 2 (8 miles) stop at gas station 4 (14 miles