> Choose just one of the following questions, write the number of your chosen question on the next page, and write your solution on the following pages. You should give an algorithm to solve the problem of your choice, find the time and space complexity of your algorithm, and briefly explain how it works. Be sure to specify the parameters you are using to measure the input size and identify the basic operation. You may tear off this first page and keep it if you want.

1. Write an algorithm that implements the word wrap operation. It should take a string and an [40 pts] integer representing the screen width. The output should be a list of strings, each representing a line of text. Make sure your algorithm doesn't split any strings and correctly handles new line characters that are encountered.

2. Design an algorithm that when given a string will output all permutations of that string. [40 pts] Find the time and space complexity of your algorithm. (For full credit your algorithm should correctly handle duplicate letters.)

3. Suppose you are investing. You want to buy low and sell high to maximize your profit. Thanks [40 pts] to the magic of time travel you have acquired an array of future prices, but can only perform two trades, one buy and one sell. Your array of prices is in chronological order and your buy order must precede your sell order. Design a $O(n \log n)$ divide and conquer algorithm for finding the maximum profit you can make.

4. Given n dice each with m faces, numbered 1 through m, find the number of ways to get a [40 pts] target sum X, where X is the summation of the values rolled on each die when all the dice are thrown. (Hint: The optimal solution involves dynamic programming and should run in $O(m \times n \times X)$ time.)

Name: _____

1. What problem did you choose _____