Problem Set 8

All parts are due on November 21, 2017 at 11:59PM. Please write your solutions in the LaTeX and Python templates provided. Aim for concise solutions; convoluted and obtuse descriptions might receive low marks, even when they are correct. Solutions should be submitted on the course website, and any code should be submitted for automated checking on alg.csail.mit.edu .

Please solve the following problems using dynamic programming.

Problem 8-1. [10 points] **Binomial Recreation**

Pascal is chilling at home doing some recreational combinatorics in his notebook, and decides he needs to compute a binomial coefficient,

$$\binom{n}{k} = \frac{n!}{(n-k)!k!}.$$

Pascal is terrible with arithmetic, so he gets out his calculator to perform some multiplications and divisions only to find that his calculator is mostly broken: only the digits and plus operator work, allowing only entry and addition of numbers. Pascal however is not deterred for he knows that binomial coefficients can also be defined recursively:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k},$$

with $\binom{n}{k} = 1$ when $k = 0$ or $k = n$. Help Pascal calculate his binomial coefficient using nothing but his broken calculator and notebook.

Problem 8-2. [15 points] **Optimal Trading**

Seorge Goros is a stock trader down on his luck. He has been consistently making the wrong trades of FredEx stock. Each day he is authorized to either buy or sell exactly s shares of FredEx at its daily price, or take no action. Ignoring inactive days, Seorge is required to alternate between buying and selling, so will never hold more than s shares between trades. To learn from his mistakes, Seorge decides to review the daily price of FredEx stock from the last d days. Describe an efficient algorithm to calculate an assignment of each day to either 'buy', 'sell', or 'none' that would have maximized his revenue during those days.

Problem 8-3. [15 points] **Constrained Gift**

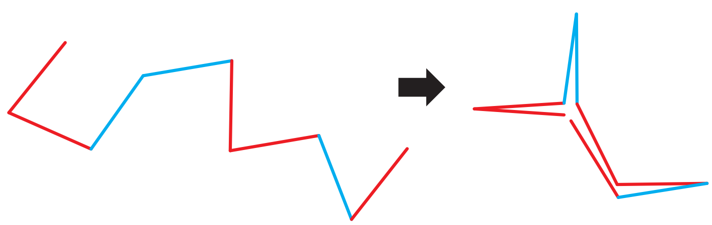
Ally Gorhythm just received a gift certificate to her favorite electronics store, though the gift certificate comes with a special condition: she must spend exactly d dollars at the store. Everything at the store is priced at a whole dollar amount, while the smallest component costs a dollar. Ally wants to spend her gift certificate by purchasing the fewest possible items at the store, allowing items to be purchased more than once. Given a price list of items in the store, describe an efficient algorithm to return a shortest shopping list to Ally.

Problem 8-4. [20 points] **Picking Teams**

A group of elementary school students take their recess very seriously. Each child has a dodge ball ability indicating how good they are at dodge ball, and every child knows every other child's ability. Every day at recess, two children are elected captains and must alternate choosing players from the remaining children to complete their teams. When picking teams the children stand in a line, and each captain takes turns adding a child to their team by choosing one of the two children at either end of the line. If you are the captain who picks first, describe a strategy to maximize the sum of dodge ball abilities on your team, assuming the other captain is also picking players to maximize her team.

Problem 8-5. [20 points] **Protein Folding**

Proteins are chains of amino acids that fold up into complex shapes. Some amino acids have natural chemical attractions to each other while others do not, and these attractions induce the protein to self assemble. Nature tends to find a folding that maximizes adjacencies between attractive pairs. Let's analyze a simplified model of this behavior. Consider a chain of n equal length edges, each edge colored either red or blue. A *valid folding* of the chain will match edges to each other to form a doubly covered tree in the plane. Describe an efficient algorithm to find a valid folding of the chain that maximizes same color matchings. An example of a valid folding that maximizes same color matchings is shown below.



Problem 8-6. [20 points] **Return of Dr. Frick**

Dr. Crancis Frick no longer thinks that permutations of a disease DNA substring in someone's DNA results in the same disease. That was a silly idea. However, he has noticed that "similar" sequences to a disease substring results in a milder form of the disease. Given two strings, the *distance* between them is the smallest number of individual insertions, deletions, or replacements of DNA letters needed to transform one string into the other. Dr. Frick observes that the severity of disease symptoms later in life is inversely proportional to the minimum distance between the disease substring and any substring of a person's DNA. For example, if the presence of disease substring (C, A, T, A, C, T) induces someone to exhibit mannerisms indistinguishable from a kitten, a human containing the substring (C, A, G, T, A, C, A) might only lead to audible purring by the individual, being a distance 2 away from the disease via removal of G and replacement of the last A by T. Given a patient's DNA, help Dr. Frick diagnose the severity of (i.e. distance to) a given disease.