# Q1. Activity-selection problem. Suppose that instead of always selecting the first activity to finish, we instead select the last activity to start that is compatible with all previously selected activities. (Ex. 16.1-2) a) Describe how this approach is a greedy algorithm [3 points]

Text, letter

Description automatically generated

# b) Prove that it yields an optimal solution [2 points]

Text, letter

Description automatically generated

# Q2. Huffman codes. What is an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers? [4 points] a:1 b:1 c:2 d:3 e:5 f:8 g:13 h:21 Can you generalize your answer to find the optimal code when the frequencies are the first n Fibonacci numbers? [3 points] (Ex. 16.3-3)

A piece of paper with writing

Description automatically generated with medium confidence

# Q3. Coin-changing problem. Consider the problem of making change for n cents using the fewest number of coins. Assume that each coin’s value is an integer. (Problem 16-1) a) Describe a greedy algorithm to make change consisting of quarters, dimes, nickels, and pennies. Prove that your algorithm yields an optimal solution. [4 points]

Text, letter

Description automatically generated

# b) Give a set of coin denominations for which the greedy algorithm does not yield an optimal solution. Your set should include a penny so that there is a solution for every value of n. [2 points]

Text, letter

Description automatically generated

# c) Write the pseudocode of the greedy algorithm for the coin-changing problem, with an amount n and coin denominations d1 > d2 > d3 > . . . > dm as its input. (Hint. You may use integer divisions in your algorithm) [3 points]

Text, letter

Description automatically generated

Text

Description automatically generated

# Q4. Rumor-spreading problem. There are n people, each in possession of a different rumor. They want to share all the rumors with each other by sending electronic messages. Assume that a sender includes all the rumors he or she knows at the time the message is sent and that a message may only have one addressee. Design a greedy algorithm that always yields the minimum number of messages they need to send to guarantee that every one of them gets all the rumors. (Hint. The minimum number of messages for n = 4 is six) [4 points]

Text, letter

Description automatically generated