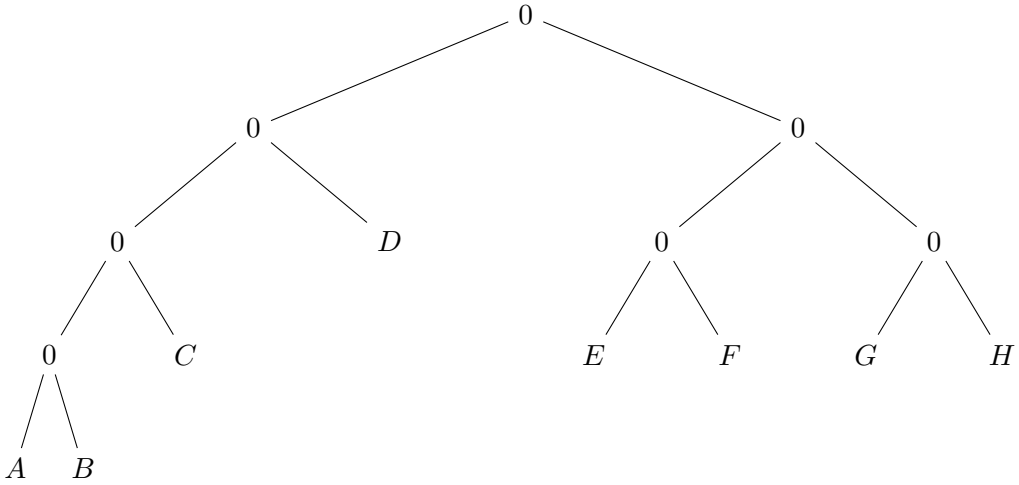
Tutorial 4, Design and Analysis of Algorithms, 2019

1. There exists a $O(n)$-time deterministic algorithm (M) for finding median of n given numbers. Using this algorithm as a subroutine, design a $O(n)$-time deterministic algorithm for solving the fractional knapsack problem (items are $(I_i)_{i=1}^n$, weight of items are $(w_i)_{i=1}^n$, profit of items are $(p_i)_{i=1}^n$, and knapsack capacity is W), and also prove its time complexity.

2. (a) Find the prefix code corresponding to the following binary tree:



   (b) Draw the binary tree corresponding to the following prefix code:
   $A = 0001, B = 0100, C = 0101, D = 0110, E = 1001, F = 1010, G = 1011, H = 1110$.

   (c) Using Huffman's algorithm find the optimal prefix code for the alphabet $\{A, B, C, D, E, F, G, H\}$ for the following frequencies:
   $f_A = \frac{1}{54}, f_B = \frac{1}{54}, f_C = \frac{2}{54}, f_D = \frac{3}{54}, f_E = \frac{5}{54}, f_F = \frac{8}{54}, f_G = \frac{13}{54}, f_H = \frac{21}{54}$.

   (d) Find *Average Bit Length* of the optimal prefix code in 2(c).

3. Show that no compression scheme can expect to compress a file of randomly chosen 8-bit characters by even a single bit.

4. (a) Find an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers:
   $a : 1$, $b : 1$, $c : 2$, $d : 3$, $e : 5$, $f : 8$, $g : 13$, $h : 21$.

   (b) Generalize your answer to find the optimal code when the frequencies are the first n Fibonacci numbers. Prove your result.

5. Suppose we have an optimal prefix code on a set $C = \{0, 1, ..., n-1\}$ of characters and we wish to transmit this code using as few bits as possible. Show how to represent any optimal prefix code on C using only $2n - 1 + n\lceil \log n \rceil$ bits. Make a binary code for the optimal prefix code of problem $2(a)$.

6. Suppose that a data file contains a sequence of 8-bit characters such that all 256 characters are about equally common: the maximum character frequency is less than twice the minimum character frequency. Prove that Huffman coding in this case is no more efficient than using an ordinary 8-bit fixed-length code.

7. Show how to transform the weight function of a weighted matroid problem, where the desired optimal solution is a *minimum-weight* maximal independent subset, to make it a standard weighted-matroid problem. Argue carefully that your transformation is correct.