Problem

LO2. Solve each of the following problems.

(a) When performing the FFT algorithm for the purpose of evaluating the polynomial

$$p(x) = -8 + 5x + 4x^2 + 6x^3 - 2x^4 + 9x^5 + 3x^6 - 5x^7,$$

at the 8th roots of unity, the algorithm recursively calls FFT on two subproblem instances. What is the polynomial for each instance. In relation to these recursive calls, why is it important that the squares of the 8th roots of unity yield the 4th roots of unity?

(b) Let (a, k) be an instance of the `Median-of-Five Find-Statistic` algorithm, where

$$a = 16, 13, 75, 76, 1, 12, 53, 13, 2, 67, 60, 35, 43, 12, 58, 35, 19, 75, 18, 62, 38, 72, 55,$$

and $k = 16$. Determine the pivot for the partitioning step when performed at the top level of recursion. After the partitioning step is completed, has the $k = 16$ statistic been found? Explain. If not provide the lower and upper indices of a for where the $k = 16$ statistic must now lie.

LO3. Solve each of the following problems.

(a) The tree below shows the state of the binary min-heap at the beginning of some round of Dijkstra's algorithm, applied to some weighted graph G. If G has directed edges

$$(a, f, 1), (d, g, 2), (f, g, 5), (f, h, 2), (f, k, 3),$$

then draw a plausible state of the heap at the end of the round.
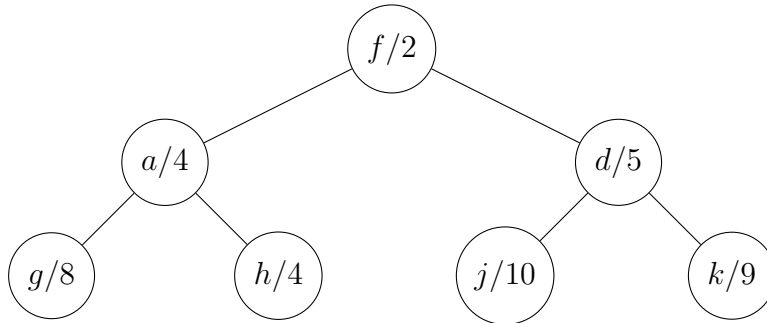
(b) Demonstrate the `Fractional Knapsack` algorithm using the input items $1 - 6$ whose respective weights are $3, 3, 4, 2, 5, 2$ and respective profits are $20, 50, 30, 10, 20, 50$. Assume a knapsack capacity equal to $M = 10$. Show all work.

LO4. Solve the following problems.

(a) The dynamic-programming algorithm that solves the `Longest Common Subsequence (LCS)` optimization problem defines a recurrence for the function $\text{lcs}(i, j)$. In words, what does $\text{lcs}(i, j)$ equal? Hint: do *not* write the recurrence (see Part b).

(b) Provide the dynamic-programming recurrence for $\mathrm{lcs}(i, j)$.

(c) Apply the recurrence from Part b to the words $u = \mathrm{baaabb}$ and $v = \mathrm{aaabbb}$. Show the matrix of subproblem solutions and use it to provide an optimal solution.

LO5. Consider the 2SAT instance

$$\mathcal{C} = \{(x_1, x_2), (\overline{x}_1, x_3), (\overline{x}_1, \overline{x}_4), (\overline{x}_1, \overline{x}_5), (x_2, \overline{x}_3), (\overline{x}_2, x_5), (\overline{x}_3, x_4), (\overline{x}_3, \overline{x}_4), (\overline{x}_3, x_6), (x_4, \overline{x}_6)\}.$$

(a) Draw the implication graph $G_{\mathcal{C}}$.

(b) Perform the Improved 2SAT algorithm by computing the necessary reachability sets. Use numerical order (in terms of the variable index) and positive literal before negative literal when choosing the reachability set to compute next. Draw the resulting reduced 2SAT instance whenever a consistent reachability set is computed. Either provide a final satisfying assignment for $\mathcal{C}$ or indicate why $\mathcal{C}$ is unsatisfiable.

(c) instance $\mathcal{C}$ of 2SAT has 400 variables and 2,498 different clauses. For the original 2SAT algorithm, what is the *least* possible number of Reachability-oracle queries that are required before one can conclude with certainty that $\mathcal{C}$ is satisfiable? Explain.

LO6. Answer the following.

(a) Provide the definition of what it means to be a mapping reduction from problem A to problem B.

(b) Given Set Partition (SP) instance $S = \{4, 6, 7, 16, 21, 25, 30, 37, 42\}$, provide $f(S)$, where $f : \mathrm{SP} \to \mathrm{SS}$ is the mapping reduction from SP to Subset Sum described in lecture.

(c) Verify that f is valid for input S from part b in the sense that both S and $f(S)$ are positive instances of their respective problems. Make sure to provide the solutions for both problem instances.