CSE 4190.101: Discrete Mathematics	Fall 2021

Problem Set 4

Instructor: Yongsoo Song **Due on:** Dec 10, 2021

Please submit your answer through eTL. It should be a single PDF file (not jpg), either typed or scanned. **Please include your student ID and name (e.g. 2020-12345 YongsooSong.pdf).** You may discuss with other students the general approach to solve the problems, but the answers should be written in your own words.

- You should cite any reference that you used, and mention what you used it for.
- The reference information should be specific so that TAs are able to find the exact material you used. For example, it is not allowed to simply mention that "I referred a lecture note of Discrete Mathematics class at * university".
- Similarly, if your reference includes a url, type it or submit a separate text file (instead of handwritten address) so that TAs can easily visit the page.
- All references should be publicly accessible. Otherwise, attach the reference to your submission.

Problem 1 (20 points)

Suppose that $\{a_n\}$ satisfies the linear nonhomogeneous recurrence relation

$$a_n = c_1 \cdot a_{n-1} + c_2 \cdot a_{n-2} + \cdots + c_k \cdot a_{n-k} + F(n),$$

where $c_1, c_2, \ldots, c_k$ are real numbers, and

$$F(n) = (b_t \cdot n^t + b_{t-1} \cdot n^{t-1} + \cdots + b_0) \cdot s^n$$

where $b_0, b_1, \ldots b_t$ and s are real numbers. Show that the following statements are true:

1. If s is not a root of the characteristic equation of the associated linear homogeneous recurrence relation, then there is a particular solution of the form

$$(p_t \cdot n^t + p_{t-1} \cdot n^{t-1} + \cdots + p_0) \cdot s^n$$

.

2. If s is a root of this characteristic equation and its multiplicity is m, then there is a particular solution of the form

$$n^m \cdot (p_t \cdot n^t + p_{t-1} \cdot n^{t-1} + \cdots + p_0) \cdot s^n$$

Problem 2 (15 points)

Suppose that each person in a group of n people votes for exactly two people from a slate of candidates to fill two positions on a committee. The top two finishers both win positions as long as each receives more than $n/2$ votes.

1. Devise a divide-and-conquer algorithm that determines whether the two candidates who received the most votes each received at least $n/2$ votes and, if so, determine who these two candidates are.
2. Use the master theorem to give a big-O estimate for the number of comparisons needed by the algorithm you devised.

Problem 3 (15 points)

Suppose someone picks a number x from a set of n numbers. A second person tries to guess the number by successively selecting subsets of the n numbers and asking the first person whether x is in each set. The first person answers either "yes" or "no". Ulam's problem, proposed by Stanislaw Ulam in 1976, asks for the number of queries required to find x, supposing that the first person is allowed to lie exactly once.

1. Show that by dividing the initial set of n elements into four parts, each with $n/4$ elements, $1/4$ of the elements can be eliminated using two queries. [Hint: Use two queries, where each of the queries asks whether the element is in the union of two of the subsets with $n/4$ elements and where one of the subsets of $n/4$ elements is used in both queries.]

2. Show that if $f(n)$ equals the number of queries used to solve Ulam's problem using the method above, then $f(n) = f(3n/4) + 2$ when n is divisible by 4. Solve the recurrence relation.

Problem 4 (10 points)

How many permutations of the 26 letters of the English alphabet do not contain any of the strings *fish*, *rat* or *bird*?

Problem 5 (20 points)

In this exercise we construct a dynamic programming algorithm for solving the problem of finding a subset S of items chosen from a set of n items where item i has a weight w_i, which is a positive integer, so that the total weight of the items in S is a maximum but does not exceed a fixed weight limit W. Let $M(j, w)$ denote the maximum total weight of the items in a subset of the first j items such that this total weight does not exceed w. This problem is known as the knapsack problem.

1. Show that
$$M(j,w) = \begin{cases} M(j-1,w) & \text{if } w_j > w, \\ \max\{M(j-1,w), w_j + M(j-1, w-w_j)\} & \text{otherwise.} \end{cases}$$

2. Construct a dynamic programming algorithm for determining the maximum total weight of items so that this total weight does not exceed W.

3. Explain how you can use the values $M(j, w)$ to find a subset of items with maximum total weight not exceeding W.

Problem 6 (10 points)

Let R be a relation on a set A and $R^{-1} = \{(b, a) \mid (a, b) \in R\}$ be the inverse relation. Show that R is antisymmetric if and only if $R \cap R^{-1}$ is a subset of the diagonal relation $\Delta = \{(a, a) \mid a \in A\}$.

Problem 7 (15 points)

Do we necessarily get an equivalence relation when we form the symmetric closure of the reflexive closure of the transitive closure of a relation?

Problem 8 (10 points)

Show that the property that a graph is bipartite is an isomorphic invariant.