CSE 113 – Data Structure and Algorithms

Assignment 1

Semester: July 2023

1. Imagine you're building a flight reservation system for an airline. The system should be able to handle the booking and cancellation of flights, as well as searching for available flights based on source, destination, and travel date. The system should also allow passengers to view their booked flights and the available seats on each flight. Create a data structure and algorithms to manage this flight reservation system effectively.
2. Consider the following sorted array of integers A = {10, 12, 14,23, 34, 56,73, 87, 89, 110, 134, 160, 187, 200}. Show the step-by-step procedure to find the target **value = 12** using binary search. Analyze the time complexity of binary search implementation and explain why it's more efficient compared to linear search for large datasets.
3. Create a structure/class for a group of 50 students holding data for their Regn no., Name, Branch, and CGPA.
   i. Apply linear search function to display data of students with a particular Regn no.
   ii. Apply bubble sort function to arrange data of students according to Regn no.
4. Consider the following sorted array of integers A = {45, 25, 56, 33, 12}. Show the step-by-step procedure the sort the data using insertion sort. Determine the total number of comparisons and swapping made by the insertion sort algorithm for the given array of integers A.
5. Programs A and B are analyzed and are found to have worst-case running times no greater than 150NlogN and N^2, respectively. Can program B run faster than program A on all possible inputs of N?
6. Solve the following recurrences
   i. T(n) = T(9n/10) + n.
   ii. T(n) = 7T(n/2) + n
   iii. (n) = 5T(n/5) + n/ lg n
   iv. T(n) = 3T(n/3 + 5) + n/2
   v. T(n) = T(n - 1) + lg n.

7. Draw the recursion tree for T (n) = 4T (⌊n/2⌋) + cn, where c is a constant, and provide a tight asymptotic bound on its solution.

8. Use a recursion tree to give an asymptotically tight solution to the recurrence T(n) = T(n - a) + T(a) + cn, where a ≥ 1 and c > 0 are constants.

9. Suppose you have an array A (-4:5, 1:9, -2:7) and the base address of A is 1000. Let's assume that each element takes 4 bytes. Now answer the following.

i. Find the length of each dimension of the array A

ii. Find the effective indices of the array A.

iii. Find the memory location of A [3, 5, 6], assume that elements are stored in row-major order.

iv. Find the memory location of A [-1, 7, 8], assume that elements are stored in column-major order.

8x6 6x4 → 8x4

10. Let A1, A2, A3, and A4 be four matrices of dimensions 8 x 7, 7 x 6, 6 x 4, and 4 x 5 respectively. Calculate the minimum number of multiplications needed to product A1A2A3A4 using Matrix Chain Multiplications (MCM).

48+30+32

11. Discuss the Rod cutting problem and draw a recursion tree for rod size 5. How dynamic programming gives an optimal solution for Rod cutting problem? Design the solution of the Rod cutting problem using dynamic programming.

12. Suppose you are given an array of size n that represents a line of units. Each unit in the array contains either a policeman ('P') or a thief ('T'). Each policeman can catch only one thief, and a policeman cannot catch a thief who is more than K units away from the policeman. Now find the maximum number of thieves that can be caught, given the array and the range K.

13. Given an integer array of coins[] = {2, 5, 3, 6} of size 4 representing different types of currency and an integer sum 10, find the number of ways to make sum by using different combinations from coins[] using dynamic programming.

14. Consider the array A[] = {12, 50, 3, 10, 7, 40, 80}. Determine the longest increasing subsequence using dynamic programming.