

Due: 12th October 2023
20% penalty for 1 day late
40% penalty for 2 days late
Submission not allowed afterwards

CS 2009: Design and Analysis of Algorithms
Assignment 2
Total Marks: 100 points

Question 1:

[10+10 = 20 Points]

Construct a $O(n \log n)$ algorithm based on the divide and conquer approach. You have to perform the following tasks for each of the problem which is given below:

- Write pseudo code or algorithm based on divide and conquer approach
- Find the recurrence relation of part (a) and then solve it by using iterative substitution method discussed in class.

Problem 1. Fast University Main Campus Karachi Cafeteria Management is decided to open a bakery with the restaurant which provides services to the students, faculty and staff based on the demand. The bakery produces or manufactures (baked) many products or item based on the needs. Suppose a list given which contains the sales of items on weekly basis and a target has been set by the management to achieve in a certain deadline of weeks. Your task is to find the 2 different and distinct weeks (not necessarily to be consecutive) whose sales is equal to the target. (where list is not necessarily sorted). your algorithm must return true if there are two distinct weeks that is $w1_sales + w2_sales = target_sales$ otherwise it returns false.

Problem 2. Upon the successful launching of Fast Main Campus bakery, the management decided to open a new branch at the Fast City Campus which provides services to the students, faculty and staff based on the demand. The bakery produces or manufactures (baked) many products or item based on the needs. Suppose a list given which contains the sales of items on weekly basis and managements again sets a target sales which they want to achieve but this time they want to know collectively contribution of both bakeries to get the sales target. Your task is to find a one unique week from each of the bakery sales list such that $w1_sales + w2_sales = target_sales$. Your algorithm must return true in this case otherwise it returns false.

Question 2:

[10+10 = 20 Points]

Suppose you are playing a story writing game and you have given a word where all the letters in the words can be considered to appear twice consecutively (e.g. xx). It can be an error if one letter appears only once. You need to design an $O(\log n)$ algorithm to find that single letter.

For the sake of simplicity assume all the words are correct and there is no need to handle any exceptional cases., i.e. the letters either appear twice or appear as a single character, but not any other case.

Also find the recurrence relation for the above problem and solve it using iterative substitution method.

Examples : Word xxyynzzwwii returns 4

: Word xxyyzzwwii returns None

: Word kklmmjjc returns 8

Question 3:

[5+10+5 = 20 Points]

Assume you have n jars, one of which is heavier (where n is a power of three). The weight of $n/3$ jars (excluding extra weight of the single jar) is given. Create an algorithm having complexity lesser than $O(n)$, to determine the heaviest jar.

- Clearly demonstrate how you could arrived at the solution.
- Write your algorithm in the form of pseudo code.
- Determine the algorithm's running time.

Question 4:

[5+5 = 10 Points]

You have studied Binary Search algorithm to search a number in an array. There are slight variants of it known as Ternary Search & Meta Binary Search. Learn about theses variants from below links;

<https://www.geeksforgeeks.org/ternary-search/>

<https://www.geeksforgeeks.org/meta-binary-search-one-sided-binary-search/>

Take an arbitrary input sorted array of size 10 (other than presented in the above links) and search a number present at the end of array using both Ternary Search & Meta Binary Search.

Show each iteration of both algorithms, Also Discuss pros and cons of both techniques as compared to original Binary Search algorithm in different scenarios.

Question 5:

[15 Points]

Describe an algorithm that, given n integers in the range 0 to k , preprocesses its input and then answers any query about how many of the n integers fall into a range $[a..b]$ in $O(1)$ time. Your algorithm should use $\Theta(n + k)$ preprocessing time.

Question 6:

[15 Points]

Show the steps by using an $O(n)$ algorithm, to sort the following list of names in ascending order. Show all the steps in each iteration.

ihab, abid, afif, fadi, adib, hadi, ibad

Hint: You do not need to show all the letters from a to z, instead you can take the last letter to be “i” for simplicity (i.e. from a to i).