DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL

Sub: Algorithms and Complexity Course Code: CS700

Max Marks: 25 Time : 11:00AM- 11-50AM

Note: Answer all the questions.

3. Solve the recurrence a) $T(n) = 3T(n/4) + n$ (using iteration method)

b) $T(n) = T(n/3) + T(2n/3) + n$ (using recursion trees) ---05 Marks

2. We are given an array of n points in the plane, and the problem is to find out the closest pair of points in the array. Derive the time complexity. ---------------- 15 Marks

3. Consider the problem of finding the k^{th} –smallest element discussed in the class. Can we divide the set into subsets of size 7 instead of 5, as done in the class. Prove or disprove $|U| < 3n/4$ in this case also --- 05 Marks

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL 222IS015

Sub: Algorithms and Complexity Class: Ist sem M.Tech Max Marks: 25

1. Prove that degree constrained spanning tree problem is NP Complete
   Instance: Graph $G = (V, E)$, positive integer $K \leq |V|$
   Question: Is there a spanning tree for G in which no vertex has degree larger than K?

   ---------------------------------- 15 Marks

2. Prove that Independent Set is ploytime reducible to circuit satisfiability.

   ----------- 10Marks

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL

Ist Sem M.Tech (CSE/CSE-IS) Mid Semester Examinations

Subject: Algorithms and Complexity

Course Code : CS700

Max Marks : 50 Date: 10-09-2022

Time: 8:30AM- 10:00AM

Note: Answer all the questions.

1. Consider the quick sort algorithm. Give the recurrence equation for the worst-case time complexity of the Quicksort algorithm for sorting n (≥2) numbers. Assume c is a constant. (02 Marks)

2. Suppose you are given an array A[1..n] of sorted integers that has been circularly shifted k positions to the right. For example, [35, 42, 5, 15, 27, 29] is a sorted array that has been circularly shifted k = 2 positions, while [27, 29, 35, 42, 5, 15] has been shifted k = 4 positions. We can obviously find the largest element in A in O(n) time. Describe an O(log n) algorithm.
(10 Marks)

3. You are given an array of positive and negative integers x[1], x[2],...,x[n]. Give an O(n log n) divide-and-conquer algorithm to find the contiguous subsequence with the largest sum. Briefly explain why your algorithm is correct.
For example, if the array was: X= { -10, 50, 60, -150, 20, 80, -10, -5, 100, -5}
Then the contiguous subsequence with the largest sum is {20, 80, -10, -5, 100}. You must use a divide and conquer strategy to get full credit. (13 Marks)

4. The edit distance problem is the minimum number of insertions, deletions, or replacements required to convert one string to another. Give a dynamic programming solution for the same. Derive its time complexity. (10 Marks)

5. State TRUE or FALSE and justify. (15 Marks)
   a) In an undirected weighted graph with distinct edge weights, both the lightest and the second lightest edge are in some MST
   b) Dijkstra's algorithm works correctly on graphs with negative-weight edges, as long as there are no negative-weight cycles.
   c) To determine whether two binary search trees on the same set of keys have identical tree structures, one could perform an inorder tree walk on both and compare the output lists.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
NITK-SURATHKAL
I[st] Sem M.Tech (CSE/CSE-IS) End Semester Examinations

Sub: Algorithms and complexity (CS 700) Max Marks : 100 Duration : 3hrs

Note : Answer all the questions. Missing data may be suitable assumed.

1. Let S be a finite set of n positive integers . You may assume that all basic arithmetic operations, i.e. addition, multiplication, and comparisons, can be done in unit time. Design an O(n log n) algorithm to verify that, if there is some subset $T \subset S$ such that the sum of the elements in T is less than $|T|^3$, then your algorithm should output "no". Otherwise it should output "yes". Prove that your algorithm is correct and analyse its running time. --------------------- 10 Marks
2. Suppose you are given an unsorted array A of all integers in the range 0 to n except for one integer, find the missing number. Assume $n = 2^k - 1$. Design an O(n) divide and conquer algorithm to find the missing number. Prove that the algorithm is correct and analyse its running time. ------------- 20 Marks
3. The **Discrete Knapsack Problem** is defined as follows. A thief is robbing a store that has n items. The i^{th} item is worth v_i dollars and weighs w_i pounds, where v_i and w_i are integers. He wants to take as valuable a load as possible, but he can carry at most W pounds in his knapsack, for some integer W. Which items should he take?

   In the **Continuous Knapsack Problem**, the setup is the same, but the thief can take fractional amounts of items, rather than having to take all or nothing of an item.

   (a) (5 Marks) Which of the two problems exhibit the optimal-substructure property? Briefly argue why or why not.

   (b) (5 Marks) Which of the two problems exhibit the greedy-choice property? Briefly argue why or why not.

   (c) (10 Marks) Describe a dynamic programming solution for the Discrete Knapsack Problem that runs in O(nW) time.
4. Prove that Independent Set decision problem is polytime reducible to Clique decision problem. ------------- 10 Marks
5. Consider the following decision problem (we'll call it the Graph Value Problem). Input: An undirected graph G with n vertices, a list L of n nonnegative integers, and a bound b.

Question: Is there a way to assign the numbers in L to the vertices of G such that (1) each number is assigned to exactly one vertex and (2) the graph's *value* is $\leq$ b? The value of the graph is the sum of all the edge values; each edge value is the product of the numbers assigned to its endpoint vertices.

a) Show that the Graph Value Problem is in NP.

b) Show that Graph-Value is NP-complete by showing that Vertex-Cover reduces to Graph-Value.

------------------------- 20 Marks

6. Give an approximation algorithm for the Metric TSP. Prove its correctness.

---------------------------------- 10 Marks

7. Explain different types of randomization algorithms with an example. Explain the randomized quick sort algorithm and derive its running time.

------------------ 10 Marks

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL

SUB: ALGORITHMS AND COMPLEXITY SURPRISE TEST : 02

MAX MARKS :25

1. Show that maximum independent set is polytime reducible to ILP -- 05 Marks

2. Show that independent set is polytime reducible to circuit satisfiability ---05 Marks

3. Prove that the Quadratic Congruences problem belongs to the class NP (Instance: Positive integers a, b, and c. Question: Is there a positive integer $x<c$ such that $x^2 \equiv a \bmod b$?) – 05

4. Prove that clique ε NPC ------------- 10 Marks

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL

I Sem M.Tech (CSE-IS) End Semester Examinations Roll No:---------------------
Date: 27-11-2024 Subject: Algorithms and Complexity Time : 9:00AM -12:00noon
Max Marks : 100 Course Code: CS700 Duration: 03 hrs.

Note: Answer all the questions. Missing data may be suitable assumed.

1. Show that Independent Set is polytime reducible to Circuit Satisfiability. Also prove that Independent Set belongs to the class NPC. -- 15 Marks

2. Prove that Formula satisfiability problem is polytime reducible to 3 CNF satisfiability problem. Also prove that Formula Satisfiability problem belongs to the class NPC. --- 15 Marks

3. Write a divide-and-conquer algorithm that finds the maximum difference between any two elements of a given array of n numbers (not necessarily distinct) in O(n) time. For example, on input $A = [4.5, 10, -2, \pi, -7.115]$, your algorithm should return 17.115. Justify briefly that your algorithm is correct and runs within the required time bound. --- 15 Marks

4. Consider the following Art Gallery Guarding problem. We are given a set $\{p_1, p_2, \ldots, p_n\}$ of positive numbers that specify the positions of paintings in a long hallway in an art gallery. We want to position guards in the hallway to protect every painting, using as few guards as possible. Suppose that a guard at location x can protect all paintings within 1 distance (i.e. in interval $[x - 1, x + 1]$), and any painting protected by at least one guard is considered protected. Give a greedy solution to the problem and prove its correctness. Derive the time complexity of your algorithm. --15 Marks

5. A new country Oddistan has a currency *oddollar* with coins of denomination 3, 5 and 7.

   (a) Can you always purchase an item worth n *oddollars*, where n >= 5, with exact change in coins. If so, explain why; otherwise, provide the smallest counterexample n > 5 (no justification is needed).

   (b) For all n that you can pay for exactly, is the combination of coins unique? If so, explain why; otherwise, provide an example n and two ways to make up *n* using the coin denominations.

(c) A new government wants to the change the coin denominations to 3, 11 and 13. Write a dynamic programming algorithm that returns (true/false) whether an exact change can be made for *n oddollars*, given any integer n > 0. Clearly define the quantity that your dynamic programming algorithm computes, identify the initial conditions, and analyze the time and space complexity of your algorithm. --- 20 Marks

6. Prove that PCS is in class NP --- 10 Marks

7. Prove that there exists a polytime algorithm that gives a set cover of size atmost *log |X| * size of optimal cover.* --- 10 Marks

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NITK-SURATHKAL

I Sem M.Tech (CSE-IS) Mid Semester Examinations.

Date : 07-10-2024 Time: 9:00AM – 10:30AM Max Marks: 50

Subject: Algorithms and Complexity

Note: Answer all the questions. Missing data may be suitably assumed.

1. Suppose you are given an unsorted array of all integers in the range to except for one integer, denoted the missing number. Assume $n= 2^k -1$. Design a O(n) Divide and Conquer algorithm to find the missing number. Argue (informally) that your algorithm is correct and analyze its running time.
----------- 10 Marks
2. Solve the following recurrences:
   a. $T(n) = 2\,T(n/2) + n/\lg n$ --------------- 05 Marks
   b. $T(n) = 4\,T(n/3) + n^{\log_3 4}$
3. There are n boxes arranged in a row. From left to right, the lengths of the boxes are $l_1, \ldots, l_n$. You want to tie up the boxes using strings. You have an unlimited supply of strings, but one string can only be used to tie up a contiguous block of boxes with total length is at most L. You want to compute an optimal solution that uses fewest possible strings to tie up all the boxes. Design an O(n) time greedy algorithm for computing a solution that uses the fewest possible strings. Argue why your algorithm runs in O(n) time. Prove that your greedy algorithm always returns an optimal solution.
---------------- 10 Marks
4. There are n dice of different colours. These are regular dice with numbers 1. . . . . 6 printed on their six sides. Given an integer m, you want to calculate the number of ways in which you can roll the dice to get the sum of numbers to be m. For example, with n = 2 dice, there are two ways of getting the sum m = 3: the two dice can roll (1, 2) or (2, 1). Note that the dice have different colours, so they are unique. Write a bottom-up dynamic program that implements dice(n, m) in $O(n \cdot m)$ time. Prove the correctness of your algorithm. -------- 10 Marks
5. A bank confiscates n bank cards suspecting them to be involved in fraud. Each card corresponds to a unique account but an account can have many cards corresponding to it. Therefore, two bank cards are equivalent if they belong to the same account. The bank has a testing machine for finding out if two cards are equivalent. They want to know that among the collection of n cards, are there more than n/2 cards that are equivalent. The only operation that the machine can do is to select two cards and tests them for equivalence. You are required to come up with a solution to this using only O(n log n) invocations of the testing machine.
-------------- 15 Marks

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NITK, Surathkal

Algorithms And Complexity (Code: CS700)
I Sem MTech. CSE
October 07, 2024

Registration No.

Time: 1:30 hour | *Mid Sem Exam* | Maximum Marks: 40

Note:- *Answer all the questions. Missing data may be suitably assumed, provided the assumption does not change the difficulty of the problem.*

1. If $f(n)$ is a k^{th} degree polynomial, then show that $f(n) = O(n^k)$. [04]

2. You are given an unsorted integer array A containing n distinct integers (assume n is a power of 2). You are supposed to design an algorithm that finds the minimum and the second minimum element in the array A. The running time for this problem is measured in terms of the number of pairwise comparisons that your algorithm performs in the worst case. (Note that you can find the minimum and the second minimum using $(2n-3)$ comparisons by a linear scan.) Design a divide-and-conquer algorithm that solves the problem using at most $(\frac{3n}{2}-2)$comparisons. [06]

3. Show how to multiply complex numbers $a+bi$ and $c+di$ using only three multiplications of real numbers. The algorithm should take a, b, c and d as input and produce the real component $ac-bd$ and the imaginary component $ad+bc$ separately. [04]

4. Let $X[1\dots n]$ and $Y[1\dots n]$ be two arrays each containing n numbers already in sorted order. Design and analyse an efficient divide and conquer $O(\log n)$-time algorithm to find the median of all $2n$ elements in arrays X and Y. [08]

5. The Discrete Knapsack Problem is defined as follows. A thief is robbing a store that has n items. The i^th item is worth v_i dollars and weighs w_i pounds, where v_i and w_i are integers. He wants to take as valuable a load as possible, but he can carry at most W pounds in his knapsack, for some integer W . Which items should he take?

   In the Continuous Knapsack Problem, the setup is the same, but the thief can take fractional amounts of items, rather than having to take all or nothing of an item.

   Which of the two problems exhibit the greedy-choice property? Briefly argue why or why not with example. [08]

6. Given n files of length $m_1, m_2, \dots\dots\dots m_n$, the optimal Tape storage problem is to find which order is the best to store them on the tape, assuming that
   i) each retrival starts with the tape rewound.
   ii)Each retrival takes time equal to the length of the preceeding files in the tape plus the length of the retrival file

   a) Describe a greedy algorithm for this problem.
   b) What is the running time of your algorithm?
   c) Prove your algorithm is correct. [10]

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

NITK, Surathkal

Algorithms And Complexity (Code: CS700)

November 22, 2024

I Semester M.Tech CSE

Registration No.

Time: 60 minutes. *Quiz-2* Maximum Marks: 20

Note:- *Answer all the questions. Missing data may be suitably assumed, provided the assumption does not change the difficulty of the problem.*

1. Given two strings $x = x_1x_2x_3...x_n$ and $y = y_1y_2....y_m$, a longest common subsequence is the largest k such that there exist indices $i_1 < i_2 < ..... < i_k$ and $j_1 < j_2 < ..... < j_k$ with $x_{i_1}x_{i_2}.....x_{i_k} = y_{j_1}y_{j_2}.....y_{j_k}$. Obtain an $O(mn)$ algorithm to find the length of the longest common subsequence in the given two input strings. (Notice that we need a subsequence here and not a substring, that is, the indices do not have to be continuous.) [08]

2. Explain Polynomial time reducibility. [02]

3. Define Class P, Class NP, Class NPC and Class NP-Hard problems. [04]

4. Give a polynomial time reduction from 3-SAT to Vertex Cover problem. Prove that Vertex Cover is NP-Hard. [06]

End of exam

MV

DEPARTMENT OF COMPUTER SCIENCE &
ENGINEERING

NITK, Surathkal

Algorithms And Complexity(Code: CS700)
I Sem MTech. CSE
September 27, 2024

Registration No.

Time: 1 hour | *Quiz - 1* | Maximum Marks: 20

Note:- *Answer all the questions. Missing data may be suitably assumed, provided the assumption does not change the difficulty of the problem.*

---

1. Solve the following recurrences. Assume that $T(n)$ is constant for sufficiently small n.

   a. $T(n) = 4T(\frac{n}{2}) + n^2\sqrt{n}$

   b. $T(n) = 4T(\sqrt{n}) + log^2 n$

   [02+02]

2. Use recursion tree to determine a good asymptotic upper bound on the recurrence $T(n) = 3T(\frac{n}{4}) + n$. Use the substitution method to verify your answer. [04]

3. Derive the best, worst and average case analysis of quick sort algorithm.

   [06]

4. In the algorithm SELECT(the k^{th} smallest in a set whose worst case linear time), the input elements are divided into groups of 5. Will the algorithm work in linear time, if they are divided into groups of 7? How about groups of 3? Analyze the running time of this algorithm?

   [06]