Indian Institute of Technology Kharagpur
Mid-Autumn Semester Examination 2022-23

Date of Examination: Sep, 2022 **Duration:** 2 Hours
Subject No.: CS21003/CS21203 **Subject:** Algorithms-I
Department/Center/School: Computer Science **Credits:** 4 **Full marks:** 50

Instructions

i. This question paper contains 2 pages and 5 questions. All questions are compulsory. Marks are indicated in parentheses. This question paper has been cross checked.
ii. Please write your name, roll number, subject name and code, date and time of examination on the answer script before attempting any solution.
iii. Use of electronic calculators only is permitted. No extra resources viz. graph papers, log-tables, trigonometric tables would be required.
iv. **Organize your work**, in a reasonably neat and coherent way. Work scattered all across the answer script without a clear ordering will receive very little marks.
v. **Mysterious or unsupported answers will not receive full marks**. A correct answer, unsupported by calculations, explanation, will receive no marks; an incorrect answer supported by substantially correct calculations and explanations may receive partial marks.

---

1. (a) Mention if the following statements are true or false and briefly explain why so.
   i. (1 point) The running time of a dynamic programming algorithm is always $\Theta(P)$ where P is the number of subproblems.
   ii. (1 point) With all equal-sized intervals, a greedy algorithm based on earliest start time will always select the maximum number of compatible intervals for the activity selection problem.
   iii. (2 points) If we use the obvious $\Theta(n^2)$ merge algorithm in the divide-and-conquer convex-hull algorithm, the overall time complexity would be $\Theta(n^2 \log n)$.
   iv. (1 point) In order to avoid re-solving subproblems repeatedly, dynamic programing does not use recursive function calls.

   (b) (1 point) Suppose $f(n) = 2n^3 + 4n + 5$ and $g(n) = 7n^5 + 5n^3 + 12$. Let $h(n)$ be an unknown function. Which of the following is **not** possible. Briefly explain your answer.
   1. $h(n)$ is $O(f(n))$ and $h(n)$ is also $O(g(n))$
   2. $h(n)$ is $O(f(n))$ and $h(n)$ is not $O(g(n))$
   3. $h(n)$ is $O(g(n))$ but $h(n)$ is not $O(f(n))$
   4. $h(n)$ is not $O(f(n))$ and $h(n)$ is also not $O(g(n))$

   (c) (4 points) Sort the functions in increasing order of asymptotic (big-O) complexity. Explain your answer.

   (1). $f_1(n) = n^{\sqrt{n}}$ (2). $f_2(n) = 2^n$ (3). $f_3(n) = n^{10} \cdot 2^{n/2}$ (4). $f_4(n) = \sum_{i=1}^{n}(i+1)$

2. (a) Find an asymptotic solution of the following recurrence. Express your answer using Θ-notation and provide some details about your calculation instead of just giving your final answer.

i. (3 points) $T(n) = 3T(n/2) + n^3$

ii. (3 points) $T(n) = 2T(n/5) + \log\log n$

(b) Consider the following two programs. For each of these programs give the asymptotic runtime as $\Theta(f(n))$ in terms of some function f and justify your work. ($x\hat{}y$ means x^y.)

i. (2 points) The first algorithm

```
Alg1(n)
    for i = 1 to n
        for j = 1 to 2^n
            Print(j);
```

ii. (2 points) The second algorithm

```
Alg2(n)
  for i = 1 to n
    for j = 1 to 2^i
      Print(j);
```

3. Indian army is planning to install an automatic monitoring system where they will put underground sensors in the border areas so that any unusual activities can be detected from the control room. The specially designed sensors have a restriction to operate properly - no two sensors can be put within the distance of 1 Km from each other (this will have interference effect). Since the border areas are large and have challenging environments, it is hard to exactly pin-point the sensor positions for the installation. However the army people have an estimate of tentative locations (GPS coordinates) where the sensors can be installed safely. But they didn't check the mutual distance restriction of 1 Km within the sensors. They only have the list of x and y coordinates of the potential locations from the GPS, and now they are trying to find out a valid set of locations from the list. Assume that the list of x and y coordinates are already sorted.

(a) (3 points) Suppose that the map is divided into a square grid, where each square has dimensions $\frac{1}{2} \times \frac{1}{2}$ Km. Now the head of the defense research predicts that they should discard all the locations which are in, or on the boundary of the same square. Do you think that this hypothesis is correct? Justify your answer.

(b) (7 points) Design an efficient algorithm to determine which location pairs are to be discarded (which are within the distance of 1 Km). Your algorithm should return such location(s) which are invalid for sensor placements. For full credit, the algorithm should run in $O(n \log n)$ time, where n is the total number of locations (*hint: think of a divide and conquer strategy that was covered in the class*).

4. (10 points) What is the LongestCommon Subsequence (LCS) of B,C,A,A,B,A,B and C,A,C,A,A,B,C? Solve using Dynamic Programming method and show the detailed calculations in a table. Highlight the cells (simply circle) along which you have got the path giving the optimal sequence.

5. Let us consider a long, quiet country road with houses scattered very sparsely along it. (We can picture the road as a long line segment, with an eastern endpoint and a western endpoint.) Further, let us suppose that despite the rural setting, the residents of all these houses are avid cell phone users. You want to place cell phone base stations at certain points along the road, so that every house is within four miles of one of the base stations.

(a) (6 points) Design an efficient *Greedy* algorithm that achieves this goal, using as few base stations as possible.

(b) (4 points) Prove that your Greedy algorithm always produces correct result.