In class Quiz 10

luo.kelin

Nov 08 2024

Note:

- Form groups of EXACTLY **two** students. You may form a group through a Piazza post.
- Discuss the **assigned** problem
  - Students with answer sheet Number $1, 5, 9, ..., 4i + 1$: Problem 1
  - Students with answer sheet Number $2, 6, 10, ..., 4i + 2$: Problem 2
  - Students with answer sheet Number $3, 7, 11, ..., 4i + 3$: Problem 3
  - Students with answer sheet Number $4, 8, 12, ..., 4i$: Problem 4

  We will review the problems in class.
- Outline your solution (an **efficient** dynamic programming algorithm) in the Answer Sheet
  - Define subproblem
  - Compute base case and recurrence relations between sub-problem
- We will distribute the answer sheets during class to each group, consisting of exactly two students seated together. Each pair of students will receive one answer sheet, which will have a designated **SheetNumber**.
- **You should submit the answer sheet paper before the end of the class. If you submitted your quiz sheet paper in class, there is no need to submit it again on UBLearns.**

  For the after class **late submission (with 50% deduction)**, both students in a group must submit the same scanned answer sheet. Please note that submissions for "In class Quiz 10 submission" on Ublearns Assignment will close **at 11:59 PM on Nov 8th.**

  If submitting after class, please scan your assigned answer sheet and name the PDF file: Quiz10_SheetNumber_FirstStudentUBIDnumber_SecondStudentUBIDnumber

  For example: Quiz10_53_12345678_12345678
- No makeup in-class quiz will be given except in provably extreme circumstances.

Problem 1: Longest Increasing Subsequence

- Input: a sequence $A = (a_1, a_2, \cdots, a_n)$ of n numbers
- Output: find a maximum-length sequence $(i_1, i_2, \cdots, i_t)$ of integers such that $1 \leq i_1 < i_2 < i_3 < \cdots < i_t \leq n$ and $a_{i_1} < a_{i_2} < a_{i_3} < \cdots < a_{i_t}$.
- Hint: Design an $O(n^2)$-time algorithm for the problem.

Problem 2: Maximum Total Weight Independent Set

- Input: a tree $G = (V, E)$ with node weights $w(v) \geq 0$ for all $v \in V$. An independent set, is a set of vertices in a graph, no two of which are adjacent.
- Output: find the independent set of the tree with the maximum total weight
- Hint: Design an $O(n)$-time algorithm for the problem, where n is the number of vertices in the tree.

Problem 3: Shortest Path With Even Number of Vertices

- Input: a directed acyclic graph $G = (V, E)$ with edge weights $w(u, v) \geq 0$ for all $(u, v) \in E$.
- Output: compute the shortest path from s to t with even number of vertices.
- Hint: Reduce the problem to the shortest path problem.

Problem 4: Counting number of inverted 5-tuples

- Input: an array A of n numbers. A 5-tuple $(i_1, i_2, i_3, i_4, i_5)$ of integers is inverted if $1 \leq i_1 < i_2 < i_3 < i_4 < i_5 \leq n$ and $A[i_1] > A[i_2] > A[i_3] > A[i_4] > A[i_5]$.
- Output: count the number of inverted 5-tuples in A.
- Hint: Design an $O(n^2)$-time algorithm for the problem