091M4041H-2019: Final Examination

Notice:

1. Please write your name along with student ID.
2. There are 6 sections in the sheet, 100 marks total.
3. When you are asked to give an algorithm, you should describe your algorithm in natural language or pseudo-codes, prove the correctness, and analyze time complexity.
4. You can write answers in either Chinese or English.

1 Divide and Conquer (16 marks)

A local minimum of an array is defined as the element which is less than its neighbors. The problem is to find one local minimum and report its index. The array may contain several local minimums, in which case you can report any one of them.
(*Hint:* All elements in the array num are distinct, and suppose that $num[-1] = num[n] = +\infty$.)

For example, in array $[2, 3, 0, 1]$, 0 is a local minimal element and your objective is to report the index number 2.

2 Dynamic Programming (16 marks)

On a staircase which has n steps, the $i - th$ step of this staircase has a non-negative cost ($cost[i]$), $(0 \leq i \leq n - 1)$. Once you pay the cost, you can either climb one or two steps.

You need to find minimum cost to reach the top of the floor, and you can either start from the step with index 0, or the step with index 1.

Please describe the optimal substructure and DP equation, prove the correctness, and analyze time complexity.

3 Greedy (16 marks)

There are n jobs $J_0, J_1, ..., J_n$, each job has a non-negative processing time p_i. There is only one server available to process these jobs, the server can't not process more than one job at once. Please schedule the jobs to minimize the average waiting time. Waiting time of J_i is the sum of processing times of all jobs scheduled before J_i.

4 Linear Programming Formulation (16 marks)

Suppose we are given n potential facility locations and a list of m clients who need to be serviced from these locations. There is a fixed cost c_j of opening a facility at location j, while there is a cost d_{ij} of serving client i from facility j. The goal is to select a set of facility locations and assign each client to one facility, while minimizing the total cost.

Please formulate this problem as an integer linear programming problem and explain the meaning of every constraint.

5 Palindromic String (16 marks)

Given a string s, we say that s is palindromic if the sequence of reading s from left to right is the same as the sequence of reading s from right to left.

Please complete the following two tasks:

1. Given a string s, please give an algorithm to find the longest palindromic subsequence's length in s.
2. What if you are asked to find the longest palindromic substring's length? Please give an algorithm to find the longest palindromic substring and its length.

Note: For an original string, a substring of it must be continuous in the original one, while a subsequence do not have to be continuous. For example, string 'bbbb' is just a palindromic subsequence of 'bbbab', while string 'bbb' is both a palindromic subsequence and a palindromic substring of 'bbbab'.

6 Integer Partition (20 marks)

An integer partition of a number n is a way of writing n as a sum of **positive** integers. Partitions that differ only in the order of their summands are considered the same.

We call an integer partition special if all its summands are **distinct**. For example, the special partitions of 6 are:

$$\begin{aligned} 6 &= 1 + 2 + 3 \\ &= 2 + 4 \\ &= 1 + 5 \\ &= 6 \end{aligned}$$

The number 6 admits many more integer partitions (a total of 11), but only those four are special.

1. Prove that, for any positive integer n, there always exists a special partition whose summands are all less than or equal to K, as long as the sum from 1 to K (including 1 and K) is greater than or equal to n. Then provide a polynomial time algorithm to construct a valid partition.
2. Given an integer n, please construct a set with the minimum number of positive integers, satisfied that all integers between 1 and n (including 1 and n) have at least one special partition whose summands are all in this set.
3. Please count the number of special integer partitions of n in polynomial time. (DO NOT count one by one as the result is extremely large)

Note: In the expression $1 + 2 = 3$, the summands are 1 and 2.