

Problem 3. (30 points)

Grade:.....

- (a) **Problem X:** Given a set S of n positive integers and another positive integer p , determine whether there exist two elements in S whose sum is exactly p .

Describe a $O(n)$ -time algorithm to solve Problem X [Hint: Hashing]:

- 1) Write down a pseudocode for your algorithm, and
- 2) explain why its worst-case asymptotic time complexity is $O(n)$.

- (b) **Problem Y:** Given a list L of n positive integers, find a subset S of these numbers such that the following two conditions hold: the total sum of the numbers in S is maximum, and S does not contain any two consecutive elements from the list (so if $L[i]$ is in S then neither $L[i - 1]$ nor $L[i + 1]$ is in S .)

Describe a $O(n)$ -time dynamic programming algorithm to solve Problem Y:

- 1) Describe your subproblems,
- 2) write a recursive formulation,
- 3) explain why the worst-case asymptotic time complexity of your algorithm is $O(n)$,
- 4) illustrate (step by step) how your algorithm runs when $L = [3, 5, 7, 8, 1]$.

No pseudocode is required.