



**United International University (UIU)**  
Dept. of Computer Science & Engineering (CSE)  
Mid Exam    Year: 2021    Trimester: Summer  
Course: CSE 2217/CSI 227 Data Structure and Algorithms II,  
Total Marks: 20, Time: 1 hour, Upload & Download: 15 min

**(Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules)**

**There are THREE questions. Answer all of them. Figures in the right-hand margin indicate full marks.**

---

1. a) What does it mean when we say that an algorithm X is asymptotically more efficient than Y? [1]
- b) Prove that divide and conquer method will give maximum sum subarray in  $O(n \log n)$  time when  $n > 1$  [2]
- c) Suppose you have to choose among two algorithms to solve a problem: [2]
- Algorithm X solves an instance of size  $n$  by recursively solving four instances of size  $n/2$ , and then combining their solutions in time  $O(n)$ .
  - Algorithm Y solves an instance of size  $n$  by recursively solving four instances of size  $3n$ , and then combining their solutions in time  $O(1)$ .
- Which one is preferable, and why?
- d) Analyze the time complexity of the following algorithm and express in big-O notation: [2]

```
func eatChips(int bowlOfChips) {  
    Println("Have some chips!")  
    for chips := 0; chips <= bowlOfChips; chips++ {  
        // dip chips  
    }  
    Println("No more chips.")  
}  
  
func pizzaDelivery(int boxesDelivered) {  
    Println("Pizza is here!")  
    for pizzaBox := 0; pizzaBox <= boxesDelivered; pizzaBox++ {  
        // open box
```

```

        for pizza := 0; pizza <= pizzaBox; pizza++ {
            // slice pizza
            for slice := 0; slice <= pizza; slice++ {
                // eat slice of pizza
            }
        }

        Println("Pizza is gone.")
    }

```

2. a) Find an optimal solution to the 0/1 knapsack instance of  $n = 4$ ,  $W = 5$ ,  $(v_1, v_2, v_3, v_4) = (50, 30, 35, 60)$ , and  $(w_1, w_2, w_3, w_4) = (2, 2, 1, 3)$ . [3+2]  
Also find the optimal solution considering the fractional knapsack problem for the same instance.

- b) Consider the following five symbols present in a file along with their frequencies: [2+1]

a	b	c	d	e
3	1	1	3	2

Construct the tree for finding codes from the Huffman encoding algorithm and write the optimal encoding codes for each symbol. **You must show the tree constructed by the algorithm and for each intermediate constructed node mention the creation sequence number.** For example, if you form a node  $t$  with  $a$  and  $b$  at first mention 1 beside the node  $t$  and then if form a node  $s$  with  $c$  and  $d$  next mention 2 beside the node  $s$ .

3. a) Discuss how the problem of computing the Fibonacci sequence demonstrates overlapping subproblems. Also, discuss how dynamic programming helps optimizing the computation. [2]
- b) How can you detect cycle in a directed graph using DFS? [1]
- c) Maloins are used as coins in the Maliceland for daily transactions. In its monetary system, Maliceland has the following coins available:  $\{1, 7, 12, 25\}$ . Show an example of an amount where the greedy strategy for the Coin-Change problem does not provide an optimal solution. [2]