Assignment Deadline 7th of December 2020

Name : ____________________
ID : ____________________
Group : ____________________

Problem 1 "Sorting":

A. Illustrate the operation of MERGE-SORT on the array A = {3, 67, 33, 89, 1, 4, 2, 23} and calculate number of comparisons.

B. What is the best and worst case complexity of MERGE-SORT Algorithm?

C. Illustrate the operation of INSERTION-SORT on the array A = {3, 67, 33, 89, 1, 4, 2, 23} and calculate number of comparisons.

D. How many number of comparisons is required for the best and worst case of INSERTION-SORT Algorithm?

E. Fill the following table with worst case complexity of each sorting algorithm.

Algorithm	Worst Case Complexity
Selection Sort	
Insertion Sort	
Bubble Sort	
Merge Sort	
Quick Sort	

Problem 2 "Asymptotic Notation":

A. For every given $f(n)$ and $g(n)$ prove that $f(n) = O(g(n))$ or $f(n) = \Omega(g(n))$ or $f(n) = \theta(g(n))$.

- $f(n) = n^5 \text{ and } g(n) = 4n^2$

- $f(n) = ln(n) \text{ and } g(n) = ln^2(n)$

- $f(n) = log(n)\ and\ g(n) = ln(n)$

- $f(n) = n^3\ and\ g(n) = log_3(n)$

B. Prove the following asymptotic notation relations:

- $T(n) = n^3 + 20n + 1\ is\ O(n^3)$

- $T(n) = n^3 + 20n\ is\ \Omega(n)$

- $T(n) = 100n^4 + 8n^2 + 5n \; is \; \theta(n^4)$

Problem 3 "Recursion Tree and Master Theorem":

A. Draw the recursion tree to find a guess to the following recurrence:

$$T(n) = \begin{cases} 2 & if \; n = 2 \\ 2T\left(\frac{n}{2}\right) & if \; n > 2 \end{cases}$$

Assume $n = 2^k, k \; is \; an \; integer$

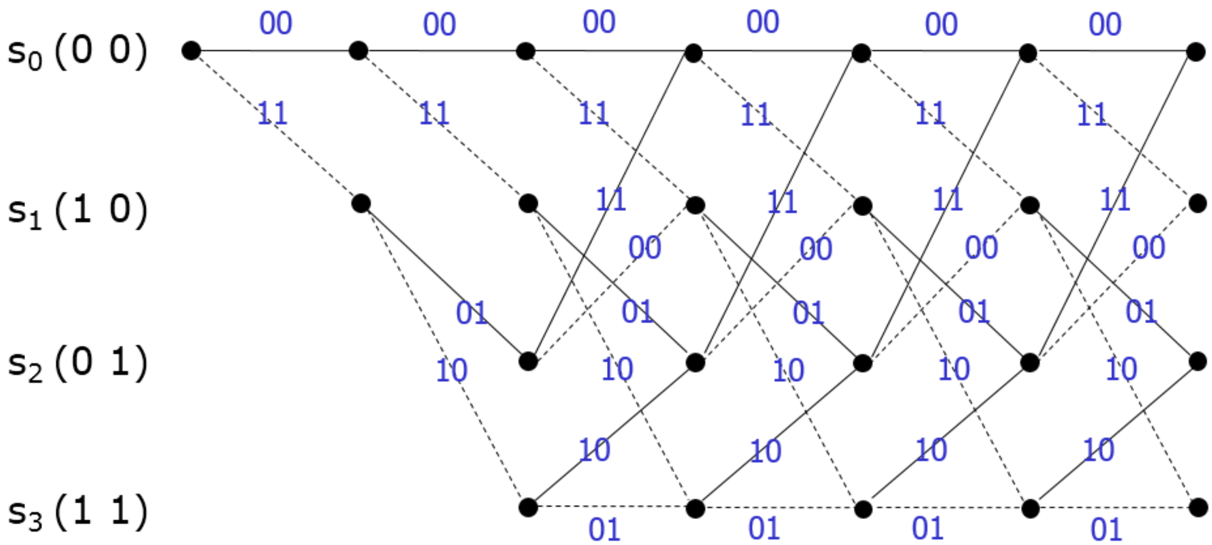
B. Present the recurrence equation of the MERGE-SORT Algorithm and solve it using the Master Theorem.

Problem 4 "Dynamic Programming":

A. VITERBI Decoding using trellis

Assume received (Hard Decision) vector is 11 10 01 00 01 10

What is the decoded vector and the message?



B. Knapsack Problem

Find the allocation using Dynamic Programming that maximizes the total benefit with constraint that the total weight of objects is less than or equal pack weight W which is 5.

Item	Weight	Benefit
1	3	100
2	2	20
3	4	60
4	1	40

Recursive formula for sub problem:

Items/Weight	w = 0	w = 1	w = 2	w = 3	w = 4	w = 5
I_1						
I_2						
I_3						
I_4						

C. Floyd Warshall

Using the Floyd-Warshall algorithm determine all pairs shortest path.