Name: _____ Roll #: _____ Section: _____

National University of Computer and Emerging Sciences, Lahore Campus

	Course:	**Design and Analysis of Algorithms**	**Course Code:**	**CS302**
	Program:	**BS(Computer Science)**	**Semester:**	**Fall 2020**
	Duration:	**90 Minutes**	**Total Marks:**	**18**
	Paper Date:	**3-Feb-21**	**Weight**	**12.5%**
	Section:	**ALL**	**Page(s):**	**5**
	Exam:	**Midterm 2**		

Instruction/Notes: Attempt the examination on the question paper and write concise answers. You can use extra sheet for rough work. Do not attach extra sheets used for rough with the question paper. Don't fill the table titled Questions/Marks.

Question	1	2	3	Total
Marks	/ 5	/5	/8	/18

Q1) In a twist to the fractional Knapsack problem, we allow the repetition of i.e. we can use any number of copies of a certain element if we wish. What will be the new greedy algorithm for this modified fractional knapsack? Give a pseudo-code. [5 Marks]

Name: _____ Roll #: _____ Section: _____

Q2) Dry run the dynamic programming algorithm to find the Max Sub-array Sum of the following input array, A:

-10	50	60	-150	20	80	-10	-5	100	-5

In array S, fill into S[i] : the optimal sum of a subarray ending at A[i]. In array P, fill into P[i]: the starting index of the optimal array ending at A[i] [5 Marks]

Solution:

S[]

P[]

Q3) Given a directed graph G (V,E), we might wish to determine whether G contains a path from i to j for all vertex pairs (i, j) ∈ V. We define the **transitive closure** of G as the graph G*=(V,E*),

where E* = {(i, j) | there is a path from vertex i to vertex j in G}.

In other words you want to find out for each vertex u, which vertices are reachable from vertex u. Give an efficient algorithm that computes the transitive closure of a given graph G(V,E). [8 Marks]

a) Briefly explain your algorithm in English.

b) Write pseudo code of your algorithm

c) Analyze its worst case time complexity.