FIT2004: Tutorial 5 (held in Week 11)
Covering concepts from Weeks 8, 9 & 10

Objectives: The tutorials, in general, give practice in problem solving, in analysis of algorithms and data-structures, and in mathematics and logic useful in the above.

Instructions to the class: Prepare your answers to the questions **before** the tutorial! It will probably not be possible to cover all questions unless the class has prepared them in advance. There is 0.5 mark worth for this Tute towards active participation. 0.25 marks is towards answering the starred questions (*) indicated below, before attending your assigned tutorial. You will have to hand in your work on these starred questions to your tutor at the very start of the tutorial. Remaining 0.25 mark is for participating during the rest of the tutorial.

Instructions to Tutors:

i. The purpose of the tutorials is not to solve the practical exercises!

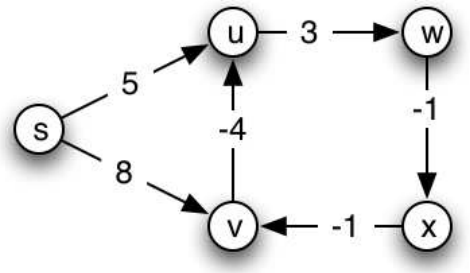
ii. The purpose is to check answers, and to discuss particular sticking points, not to simply make answers available.

1. *

   (a) Revise the entire Bellman-Ford's algorithm given in week 9 lecture slides (see slide 11)

   (b) On slide 11, you will notice that Bellman-Ford's algorithm has a postprocessing step to ensure that no negative-weight-cycle was encountered along a path to some vertex in the graph from the source. The criteria used to detect such cycles is given in STEP 3. Reason why this criteria is correct.

   (c) Work out the algorithm on the following graph by filling the table with your calculations as you proceed through the algorithm.



	i=0	i=1	i=2	i=3	i=4
s	0				
u	∞				
v	∞				
w	∞				
x	∞				

2. Work out the time complexity of Kruskal's algorithm for Minimum Spanning Tree problem using the Union-Find data structure introduced in Week 10 lecture.

3. Reason why kruskal's algorithm is correct.

4. Below is a subset of units some of you are learning with their associated prerequisites (in a simplified way).

   FIT1040 (Programming Fundamentals) no prerequisites

   MAT1830 (Discrete Math for CS) no prerequisites

   FIT3042 (Systems Tools and Programming Languages) FIT1008

   FIT1008 (Introduction to CS) FIT1040,FIT1002,FIT1029

   FIT2004 (Algorithms and DS) FIT1008,MAT1830

   FIT1029 (Algorithmic Problem Solving) no prerequisites

   FIT2014 (Theory of Computation) FIT1029,MAT1830

   FIT3036 (CS Project) FIT2004, FIT3140

   FIT1002 (Computer Programming) no prerequisites

   FIT3140 (Advanced Programming) FIT1008

   MAT2003 (Continuous Math for CS) MAT1830

   FIT3139 (Computational Science) FIT1002, FIT1040,MAT1830,MAT2003

   Draw a graph and run topological sort on this graph. (This ought to give you an ordering of units that some of you are taking along your pathway towards a Bachelor's degree.)

-=END=-