The University of New South Wales - COMP9312 - 24T2 - **Data Analytics for Graphs** - Exam Sample

Exam Sample Questions

Summary

Submission	Submit an electronic copy of all answers on Moodle (Only the last submission will be used).
Required Files	Only `.pdf` file is accepted. The file name should be `exam_Zid.pdf`
Duration	**10am--1pm Monday 19 August**
Marks	**100** marks (**50%** toward your total mark for this course)

By attending this exam, you consent to the following policy:
I acknowledge that all of the work I submit for this exam will be completed by me without assistance from anyone else. I will not copy the questions to any site outside CSE except onto my home computer.

No late penalty policy is applied. Anything submitted after **1pm Monday 19 Aug** will be ignored (unless you have adjustments based on an Equitable Learning Plan).

We will run **plagiarism-checking** on your submissions. Plagiarism, especially in exams, will be prosecuted as Student Misconduct.

You might want to keep this page open in a browser tab in case you need to refer to it. You should also have a **mail reader** open to receive any major updates on the exam (which will also be posted on the Webcms3 Notices).

To ask for clarification on any question, create **a private post** on the EdForum. Do not public any answer on the Forum.

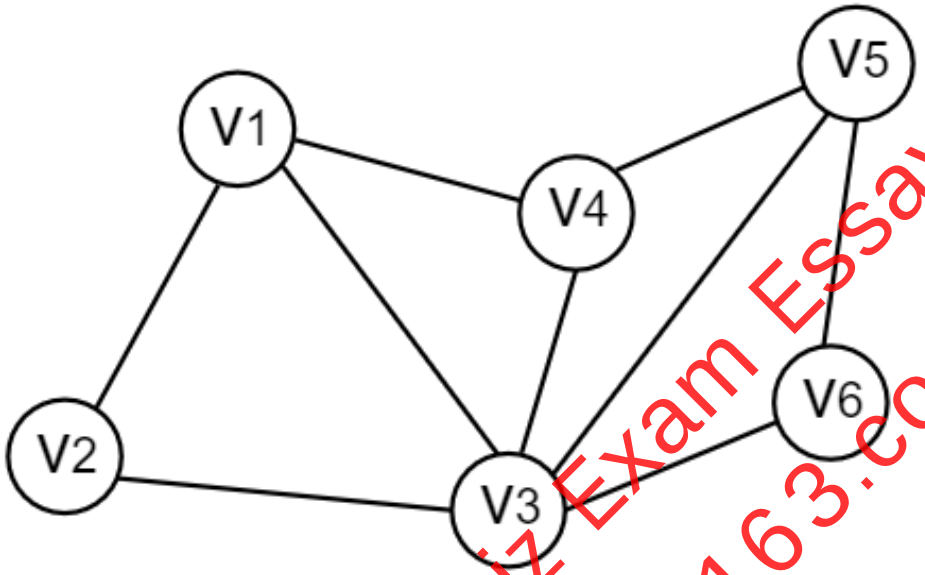
General Instructions:

- Questions are not worth equal marks.
- Questions may be answered in any order.
- **Do not leave submissions to the last minute.**
- Check that you submitted all of your answers.

If you believe that insufficient information has been provided to answer a given question, then you should write any assumptions that you think are necessary to complete the question and continue work from there. If the assumptions are reasonable, you can still obtain full marks for the question.

START OF QUESTIONS

Q1



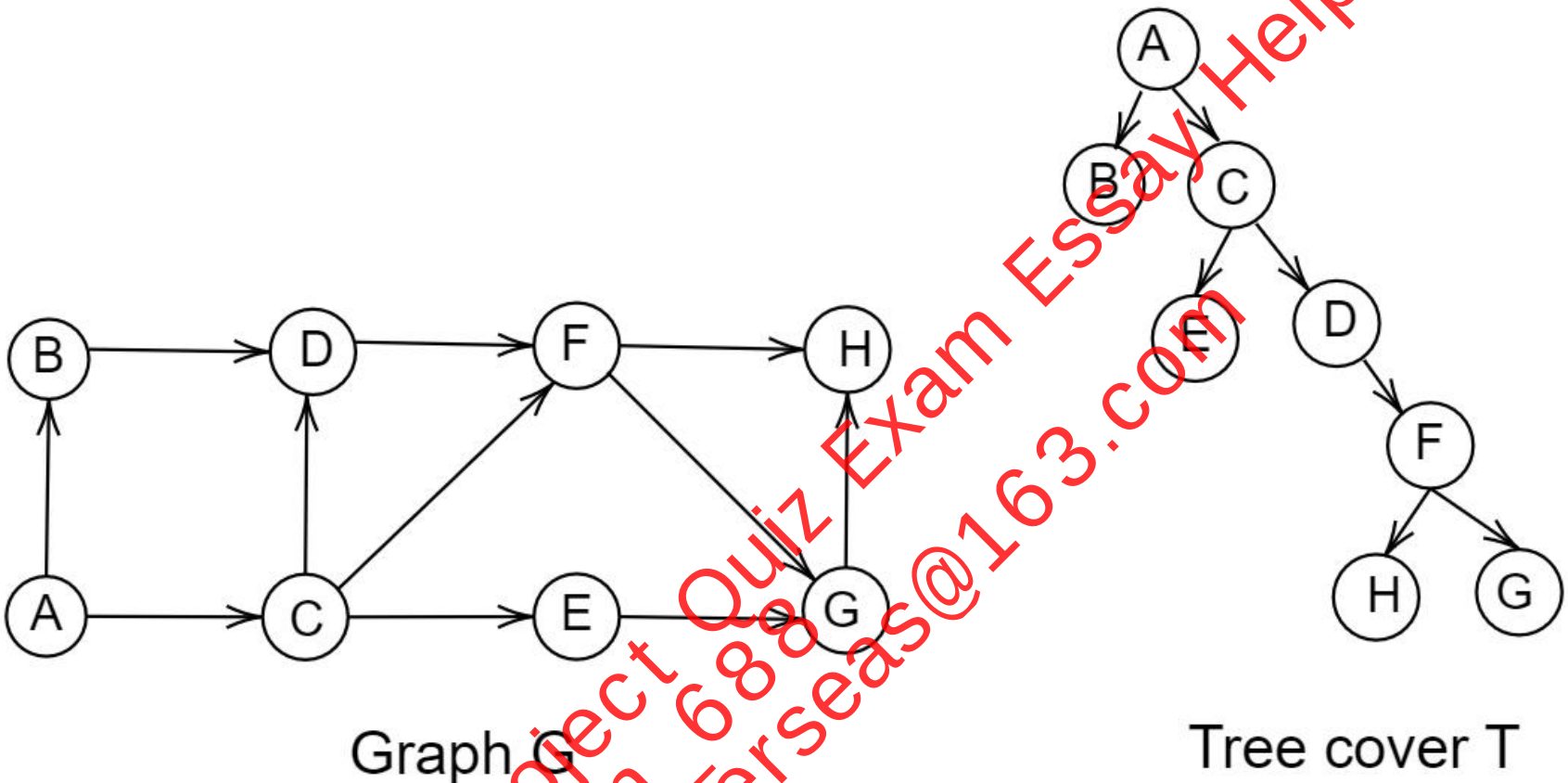
Consider the above graph and answer the following questions. Please show necessary intermediate steps.

1.1 Calculate the density of the graph.

1.2 Please use the Compressed Sparse Row (CSR) to represent the above graph.

1.3 Commonly used data structures for graph stroage include adjacency matrix, adjacency list and CSR. Assume that we are dealing with a static graph and we aim to run the Pagerank algorithm for the graph. Which data stucture is the best to represent a graph in the above case? What scenarios the other two data structures are more suitable in, repsectively? Justify your answer.

Q2



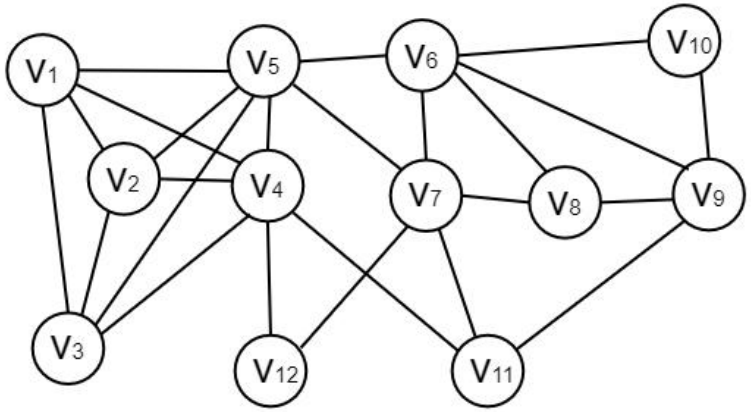
Consider the above directed acyclic graph G and answer the following questions. Please show necessary intermediate steps.

3.1 Present the transitive closure for the given DAG.

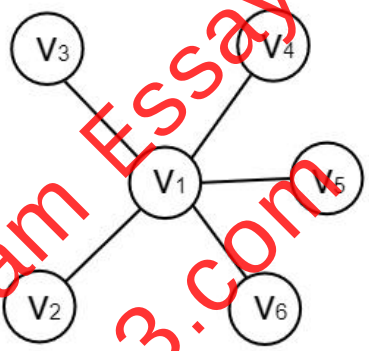
3.2 Given the tree cover T, construct the corresponding compression scheme (do not merge adjacent intervals). Report the number of intervals used in the final compression scheme.

3.3 Show the result of the total-order-based 2-hop cover of the graph. Note that you should show your node order.

Q3



Graph G1



Core(v_2)=2
Core(v_3)=3
Core(v_4)=4
Core(v_5)=5
Core(v_6)=6

Graph G2

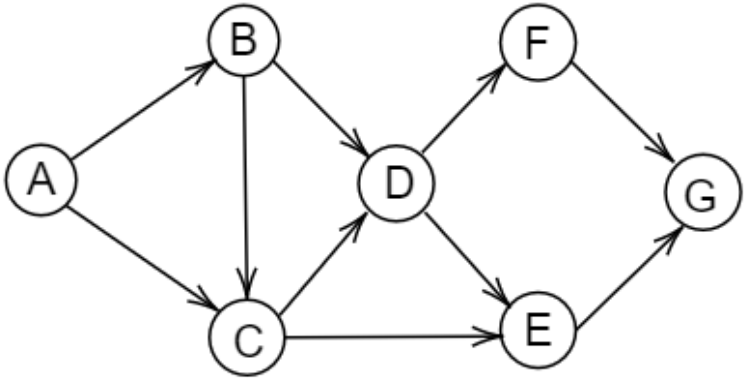
Consider the above graphs and answer the following questions. Please show necessary intermediate steps.

3.1 Does this graph G1 contains 4-core? If your answer is "yes", please find the 4-core in this graph. If your answer is "no", please explain why.

3.2 Does this graph G1 contains 5-truss? If your answer is "yes", please find all edges in 5-truss in this graph. If your answer is "no", please exaplain why.

3.3 The graph G2 is a subgraph from a large-scale graph. Node v1 has 5 neighbors, and we know their core numbers, which are shown in the figure. Please calculate the core number of node v1.

Q4



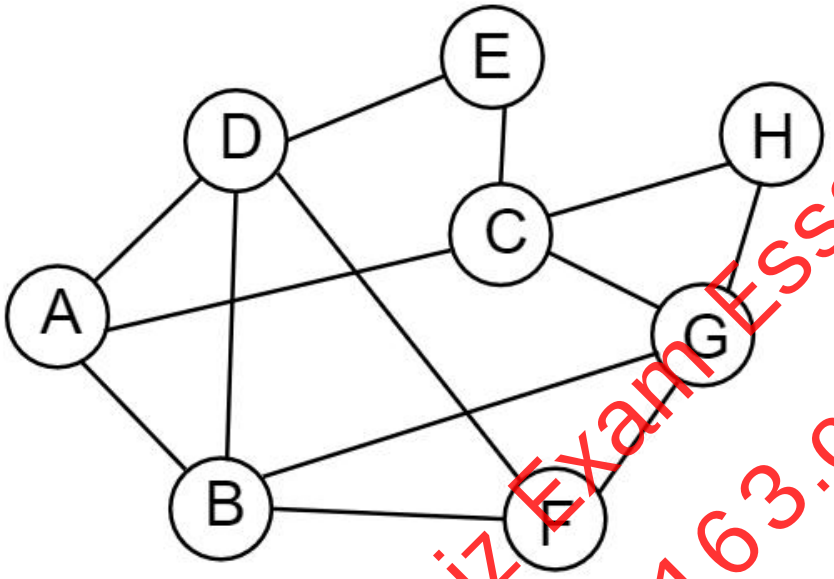
$$PR(A) = \frac{1-d}{N} + d\left(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \cdots + \frac{PR(T_n)}{C(T_n)}\right)$$

Consider the above graph and answer the following questions. The PageRank formulation is given as above where A is a given page, T_i is the page that points to page A, d is the damping factor and is set as 0.85 in this question, C(T) is the number of links going out of T, N is the number of pages in the graph, PR(A) is the PageRank value of page A (initialized as 1/N for each page).

4.1 Please compute the PageRank value of page D of the above graph after one iteration.

4.2 Assume that we aim to implement pagerank algorithm using Pregel in distributed environment. We have three machines X, Y, and Z. We know that nodes A and B are located in X. Nodes C and D are located in Y. Nodes E, F, and G are located in Z. We run the PageRank algorithm in one iteration. How many messages are generated if the combiner is used? How many messages are generated if the combiner is not used? Justify your answer.

Q5



Graph G6

Consider the graph G6. The node embeddings for all nodes are stacked in H. We use the Dice-Similarity which is defined as follows to measure the similarity between two vectors $u, v \in R^k$.

$$Dice(u, v) = \frac{2\sum_{i=1}^{k} u_i v_i}{\sum_{i=1}^{k} \left(u_i^2 + v_i^2\right)}$$

5.1 Based on the above similarity function, which node has the highest similarity with node C. Which node has the lowest similarity with node C. Justify your answer.

5.2 Complete the table below. Use the symbol "T" the to indicate that the node embedding method uses the corresponding information to learn the node embedding. Otherwise, input the symbol "F".

5.3 Which algorithm has a better performance in an inductive learning manner, GraphSAGE or GCN? Give an explanation about your choice.

$$H = \begin{vmatrix} 0.73 & 0.59 & 0.46 & 0.14 \\ 0.73 & 0.12 & 0.31 & 0.68 \\ 0.64 & 0.19 & 0.40 & 0.62 \\ 0.22 & 0.29 & 0.55 & 0.72 \\ 0.04 & 0.51 & 0.80 & 0.73 \\ 0.48 & 0.34 & 0.46 & 0.50 \\ 0.50 & 0.48 & 0.71 & 0.11 \\ 0.31 & 0.72 & 0.32 & 0.51 \end{vmatrix}$$

Method	Structure Information	Label Information	Feature Information
Node2Vec			
GCN			
GraphSAGE			
GAT			

Q6

6.1 Which of the following technique is adopted by GraphSAGE but not adopted by GCN? (only one correct choice)

a. Aggregation of messages from neighbor nodes.
b. Adopting multiple GNN layers to be more expressive.
c. Sampling a subset of neighbor nodes for each node to reduce computation.
d. Need activation function to add nonlinearity.

6.2 Please determine whether the following statements about the benefits of attention mechanism used in graph neural networks are TRUE or FALSE.

a. The computation of attentional coefficients can be parallelized across all edges of the graph.

b. It allows different importance values of different neighbors for each node and the mean of importance values for each node retain the same.
c. It has fixed number of parameters, irrespective of graph size.
d. It does not depend on the global graph structure so that has inductive capability.

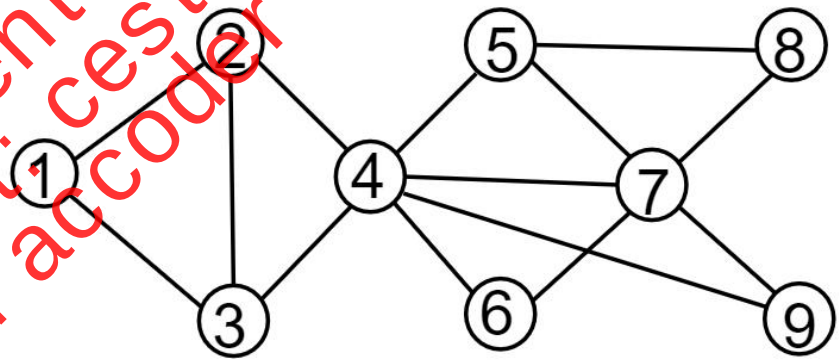
6.3 Please determine whether the following statements about Graph Attention Network are TRUE or FALSE.

a. It adopts the DeepWalk model to assign an attention value for each node.
b. It can be used for the node classification task.
c. It does not sample neighbor nodes for each node but aggregate all the neighborhood messages.
d. It is flexible to adopt multi-head attention.

6.4 Give the matrix formulation of GCN as below, why do we multiply $H^{(l)}$ and B_i^T ? Why do we multiply D^{-1} and A ? Give a short explanation.

$$H^{(l+1)} = \sigma\left(\widetilde{A}H^{(l)}W_i^T + H^{(l)}B_i^T\right) where\ \widetilde{A} = D^{-1}A$$

Q7



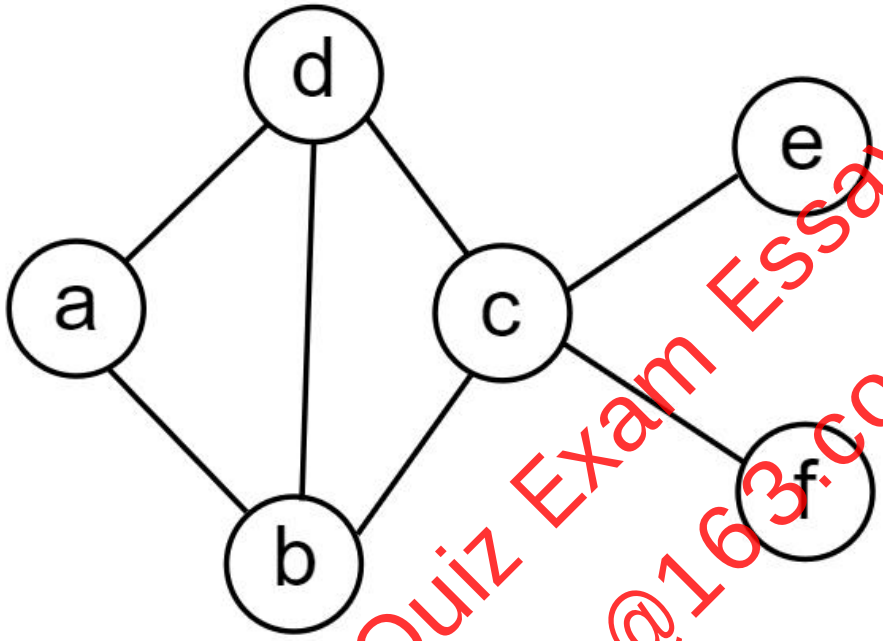
Consider the GAT model on the above undirected graph with 9 nodes. Please compute the output of the first graph convolutional layer (i.e.,H^1) step by step base on the following formula:

$$h_v^{(l)} = \sigma\left(\sum_{u \in N(v)} \alpha_{vu} W^{(l)} h_u^{(l-1)}\right)$$

where h_v^l indicates the d_l-dimensional embedding of node v in layer l, and $H^l = [h_{v1}^l, h_{v2}^l, \cdots, h_{v9}^l]$. $\alpha_{vu} = \frac{1}{N(v)}$ is the weighting factor of node u's message to node v. $W^l \in R^{d_{l+1} * d_l}$ denotes the weight matrix for the neighbors of v in layer l, d_l denotes the dimension of the node embedding in layer l, $N(v)$ denotes the neighbor set of the node v, $B^l \in R^{d_{l+1} * d_l}$ denotes the self-looping weight matrix in layer l, $\sigma(\cdot)$ denotes the ReLU non-linear function. The initial embeddings for all nodes are stacked in H^0. W^1 is the weight matrices of layer 0.

$$H^0 = \begin{vmatrix} 0.30 & -0.60 & 0.10 & 0.20 \\ 0.60 & 0.30 & 0.40 & -0.10 \\ 0.20 & 0.70 & -0.40 & 0.50 \\ -0.40 & 0.60 & 0.10 & 0.80 \\ 0.40 & 0.90 & -0.20 & 0.10 \\ 0.30 & -0.30 & 0.90 & 0.70 \\ -0.20 & 0.20 & 0.70 & 0.40 \\ -0.60 & 0.50 & 0.10 & 0.70 \\ 0.20 & 0.60 & 0.70 & -0.10 \end{vmatrix} \qquad W^1 = \begin{vmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix}$$

Q8



Consider the above graph and answer the following questions. Please show necessary intermediate steps.

8.1 Please determine whether the following statements are TRUE or FALSE, and justify your answer.

a. In any correct DFS traversal starting from a, c must be traversed after b.
b. In some correct BFS traversal starting from c, a can be traversed before e.
c. In some correct DFS traversal starting from d, a can be traversed before e.
d. In any correct BFS traversal starting from b, e must be traversed after d.

8.2 Given the following pseudo-code of the prim's algorithm to compute the minimum spinning tree of an undirected weighted connected graph. Please analyze its time complexity, and present how to improve its time complexity (show how to revise the pseudo-code).

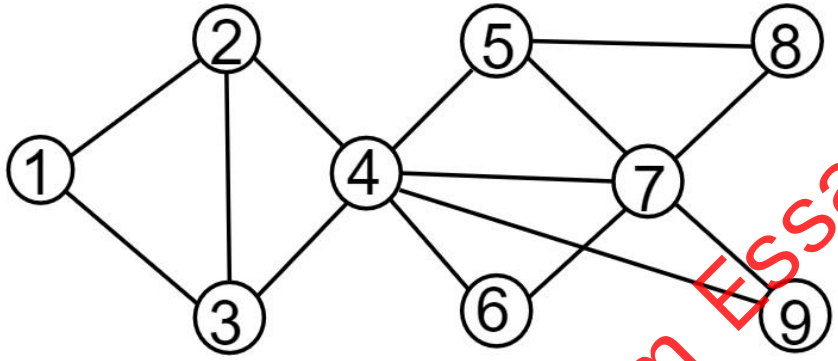
```
function MST-Prim(G(V,E)):
Input: An undirected weighted graph G
```

Output: Edges in the MST T

```
T = an empty list;
for each node v:
    d[v] = ∞, p[v] = NULL, visited[v] = false;
pick an arbitrary node u, d[u] = 0, visited[u] = true;
add (d[u],u) to a priority queue Q;

While Q is not empty:
    v = pop the node with the smallest d[] value from Q;
    if visited[v]: continue;
    visited[v] = true;
    if p[v] != NULL:
        add the pair (v,p[v]) to T;
    for each neighbor u of v:
        if !visited[u] && weight(u,v) < d[u]:
            d[u] = weight(u,v);
            add (d[u],u) to Q;
            p[u] = v;
return T;
```

Q9



9.1 A cycle is a simple path in which the terminals are the same. For instance of the graph in Q9, {1,2,4,3,1} is a cycle. Assume that we have an undirected graph G(V,E) and a sequence of new edges inserted to G. For each new edge e, please design an algorithm to test if there exists a cycle containing e in the graph. Note that you are expected to preprocess the graph and maintain some index (data structure) to check cycles efficiently.

9.2 A clique is a complete graph in which every pair of nodes are connected. Given an integer k, a k-clique is a clique with k nodes. Given an undirected graph G(V,E) and an integer k, please write the pseudo-code to enumerate all k-cliques in G, and analyze the time complexity of your algorithm.

END OF QUESTIONS