

This assignment, along with other assignments will carry 5% of your final grade.

Submission Deadline: 11:59PM, 20 November, 2023

You have to submit a soft copy of the assignment.

Softcopy submission:

- Please write your answers in A4 papers and scan your answer script.
- Rename the file with your student id (e.g "21201123.pdf").
- Submit the pdf file.
- Maximum file size is 10MB.
- The filename should contain your student id only.
- **Failing to follow the submission format will lead to a deduction in marks.**
- **Any sort of plagiarism will not only result in 0 in the assignment but will also result in a 0 in class performance.**
- **Submission link:**
 - Section-02: <https://forms.gle/oaSgvjTYprDSh3W87>
 - Section-03: <https://forms.gle/EtVj8HS69tu3vxSm8>

CSE-221 (Algorithms)

Assignment 02

1. Calculate the time complexity of the following recurrence relations.
[Any method is acceptable as long as steps are shown]

- A. $T(n) = 2T(n/2) + 1/n$
- B. $T(n) = 2T(n/3) + n$
- C. $T(n) = T(n/2) + T(n/5) + n$
- D. $T(n) = 2T(n/4) + n^2$

2. You have been asked to sort the following array of integers in ascending order.

Index	0	1	2	3	4	5	6	7
Number	23	21	19	15	12	11	5	3

You decided to use Quick sort using the first element as pivot for the task. However your teacher says “ Your approach won’t be efficient in this case”.

- A. Why do you think your teacher says so?
- B. Write the recurrence relation of your approach and calculate the time complexity. You have to show the steps and proper mathematical logic.

3. Inspired by the Karatsuba algorithm, a curious CSE student Benjamin decided to modify the algorithm in his own way. For a n digit number, instead of using subproblems of size $n/2$, Benjamin used subproblems of size $n/3$. He believes with this modification, he can get a faster algorithm than Karatsuba’s. So, for finding product of two n digit numbers A and B , Benjamin splitted A into 3 subproblems (A_1, A_2, A_3) each with $n/3$ digits and B into 3 subproblems (B_1, B_2, B_3) each with $n/3$ digits. Benjamin wants to write a divide and conquer algorithm for finding the product of A and B from these smaller subproblems. (Assume n is a power of 3)

- A. Write A in terms of A_1, A_2, A_3 and n . Write B in terms of B_1, B_2, B_3 and n
- B. Calculate the product AB from your answers in (A)
- C. Help Benjamin write the pseudocode of the divide and conquer algorithm
- D. Calculate the time complexity of your algorithm and validate whether Benjamin’s claim of getting a faster algorithm is true or not.

4. Suppose we have the following social network, Friendsbook where:

Person A is friends with Person B, C, D, and E.
Person B is friends with Person A, C, and F.
Person C is friends with Person A, B, D, and G.

Person D is friends with Person A, C, E, G, and H.

Person E is friends with Person A, D, and H.

Person F is friends with Person B and G.

Person G is friends with Person C, D, F, and H.

Person H is friends with Person D, E, and G.

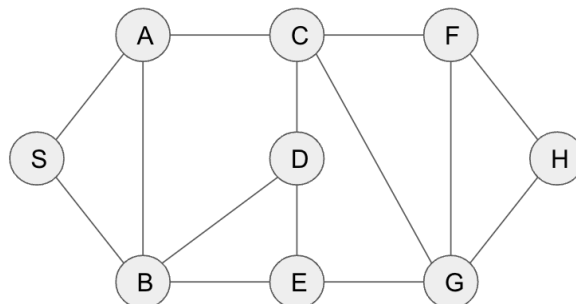
- A. Draw a graph to represent the network using each user as a node or vertex and the friendships as edges between the nodes.
- B. Create the adjacency list and adjacency matrix for the graph representation
- C. For each pair of users, calculate the number of friends they have in mutual.
- D. In Friendsbook, a user can share posts. Suppose the CEO of Friendsbook hired you for developing their post display system. You were given the task to implement a system that displays posts from users who are at most two degrees of connection away from the one who is scrolling his feed. After looking at the requirements you came up with an idea to implement the system using the BFS algorithm. Now, for person F, simulate your algorithm. Can F see the posts of all other users in his feed?

5. The following information is stored after implementing the DFS algorithm on an undirected graph.

Nodes	1	2	3	4	5	6	7
Parent							
Starting Time	8	11	5	6	4	1	2
Finish Time	9	12	10	7	13	14	3
Distance from Root							

- A. Draw the DFS tree.
- B. Fill up the empty rows of the table.

6. Consider this graph:



- A. Justify $\sum_{v \in V} \deg(v) = 2m$, where m = number of edges in the graph.
- B. Calculate the number of additional edges that can be added between the nodes randomly, keeping the graph 'simple' (without adding multiple edges between any two nodes, loops in a single node).