BACHELOR OF COMPUTER APPLICATIONS (BCAOL)

(Revised Syllabus)

BCA(Revised Syllabus)/ASSIGN/SEMESTER-IV

ASSIGNMENTS

(July – 2025 & January – 2026 sessions)

(BCS-040, MCS-024, BCS-041, BCS-042,

MCSL-016, BCSL-043, BCSL-044, BCSL-045)

SCHOOL OF COMPUTER AND INFORMATION SCIENCES
INDIRA GANDHI NATIONAL OPEN UNIVERSITY
MAIDAN GARHI, NEW DELHI – 110 068

CONTENTS

Important Notes

1. Submit your assignments to the Coordinator of your Study Centre on or before the due date.
2. Assignment submission before due dates is compulsory to become eligible for appearing in corresponding Term End Examinations. For further details, please refer to BCA Programme Guide.
3. To become eligible for appearing the Term End Practical Examination for the lab courses, it is essential to fulfill the minimum attendance requirements as well as submission of assignments (on or before the due date). For further details, please refer to the BCA Programme Guide.

Course Code	**:**	**BCS-042**
Course Title	**:**	**Introduction to Algorithm design**
Assignment Number	**:**	**BCA(IV)/042/Assignment/2025-26**
Maximum Marks	**:**	**100**
Weightage	**:**	**30%**
Last date of Submission	**:**	**31st October, 2025 (For July Session)**
	:	**30th April, 2026 (For January Session)**

This assignment has 8 questions of 10 Marks each, answer all questions. Rest 20 marks are for viva voce. Please go through the guidelines regarding assignments given in the Programme Guide for the format of presentation.

Q1. Explain the following fundamental techniques, used to design an algorithm efficiently:

- Divide-and-Conquer
- Greedy method
- Dynamic Programming
- Backtracking
- Branch-and-Bound

Q2. Prove the following proposition using induction:

P(n): $1^1 + 2^2 + 3^2 + 4^2 + \cdots \ldots \ldots \ldots \ldots \ldots + n^2 = \frac{n(n+1)(2n+1)}{6}$

Q3. For the function defined by $f(n) = 2n^3 + 3n^2 + 1$ and $g(n) = 2n^2 + 3$, show that

(i) $f(n) = \Omega(g(n))$ (ii) $g(n) != \Omega(f(n))$ (iii) $n^3 = \Omega(g(n))$ (iv) $f(n) != \Omega(n^4)$

Q4. Solve the following recurrence Using Recursion tree method

(i) $T(n) = 3T(\frac{n}{2}) + n$

(ii) $T(n) = 2T(\frac{n}{2}) + n^2$

(iii) $T(n) = T(\frac{n}{2}) + T\left(\frac{n}{4}\right) + T\left(\frac{n}{8}\right) + n$

Q5. Analyze best case, average case, and worst-case time complexities of following algorithms with the help of suitable examples.

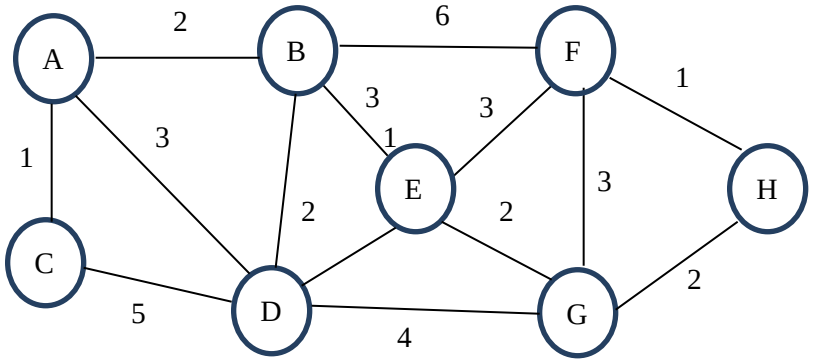
(i) Insertion sort

(ii) Binary sort

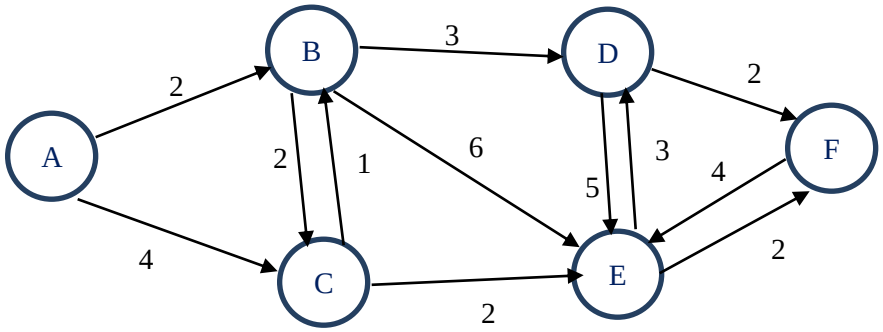
(iii) Binary search

(iv) Merge sort

Q6. Apply Kruskal's Algorithm on the following graph to find minimum cost spanning tree



Q7. Apply Dijkastra's Algorithm to find the shortest path from source vertex 'A' to all other vertices for following graph.



Q8. Explain DFS and BDS Graph traversal algorithms with the help of a suitable example.