GOVERNMENT POLLYTECHNIC, NAYAGARH
PREPARED BY :- TANMAY NATH MISHRA
SUB :- ALGORITHM

UNIT–1: Introduction to Algorithm

2 Marks Questions

1. What is an algorithm?
2. What is the significance of analyzing algorithms?
3. What are characteristics of a good algorithm?
4. Differentiate between iterative and recursive algorithm.
5. What is pseudo code?
6. Differentiate between algorithm and pseudo code.
7. What is a program?
8. Write the difference between algorithm and program.
9. What are the types of algorithm efficiencies?
10. What is pseudo code, and why is it useful in algorithm design?

5 Marks Questions

1. What are the fundamental steps involved in algorithmic problem solving?
2. Define algorithm. Write down the different criteria for a good algorithm.
3. Illustrate an algorithm for finding sum of natural numbers.
4. Write an algorithm to find the sum of digits of a number.
5. Discuss the relationship between algorithms and programs.

10 Marks Questions

1. Define an algorithm. List the different criteria of algorithm. Discuss the fundamental steps involved in algorithm.

2. Differentiate between algorithm and pseudo code. Write an algorithm to find the sum of all odd numbers between 1 and 100.

UNIT–2: Algorithmic Complexity

2 Marks Questions

1. Define asymptotic notation.
2. What is order of growth?
3. Define O-notation.
4. Define Ω-notation.
5. Define θ-notation.
6. What are the steps involved in the analysis framework?
7. What is the need for analyzing algorithms?
8. Define time complexity.
9. Define space complexity.
10. Define best case and worst case complexity.

5 Marks Questions

1. Discuss factors affecting the time complexity.
2. Analyze space complexity and time complexity with example.
3. Compare between a priori analysis and a posteriori analysis.
4. What are the different cases of time complexity analysis? Explain with examples.
5. Explain Big-O notation with an example.

10 Marks Questions

1. What is asymptotic notation? Explain different types with examples.
2. Discuss briefly about Big O, Big Omega and Theta notation.

UNIT–3: Recursive Algorithms

2 Marks Questions

1. Define recursive algorithm.
2. Define non-recursive algorithm.
3. What is a base case?
4. How does the call stack work in recursion?
5. What happens if there is no base case?
6. How do you distinguish between iteration and recursion?
7. What are the essential components of a recursive algorithm?
8. What are the limitations of recursion?
9. What are direct and indirect recursion algorithms?
10. What are the advantages of recursion?

5 Marks Questions

1. What are some potential problems with recursive solutions?
2. Differentiate between recursive and iterative algorithm with example.
3. Write an algorithm to print Fibonacci series up to n terms.
4. Write an algorithm to find factorial using recursion.

5. What is recursion? Write advantages and disadvantages.

10 Marks Questions

1. Differentiate between recursive and iterative algorithm with example. What are some potential problems with recursive solutions?
2. What is recursion? Write advantages and disadvantages. Write algorithm for factorial of a number.

UNIT–4: Algorithm Paradigms

2 Marks Questions

1. What is the Greedy algorithm?
2. Define Divide and Conquer approach.
3. What is Dynamic Programming?
4. What is Backtracking in algorithm design?
5. What is the activity selection problem and how can it be solved using a greedy approach?
6. What is greedy algorithm used for?
7. How does dynamic programming help solve knapsack problem efficiently?
8. What are the difference between top-down and buttom-up dynamic programming?
9. Can you explain the concept of backtracking and how it relates to recursion?
10. What are the main challenges or limitations associated with Backtracking algorithms?

5 Marks Questions

1. Describe Divide and Conquer with example. How is it different from greedy approach?
2. Differentiate between Dynamic Programming and Greedy approach.

3. Explain Longest Common Subsequence with example.

4. Explain the Greedy algorithm paradigm with an example. How does it work?

5. Explain Dynamic Programming with example. How is it different from Divide and Conquer?

10 Marks Questions

1. Explain how branch and bound algorithms work. What are the advantages of Branch and Bound approach? Write down the common applications of branch and bound approach. .

2. Explain Greedy algorithm for Huffman Coding.

UNIT–5: Sorting and Searching

2 Marks Questions

1. What is meant by sorting and searching?

2. What is linear search?

3. What is binary search?

4. Explain the working principle of Bubble Sort.

5. Explain the basic idea of Selection Sort.

6. What is a Symbol Table in searching?
7. Define Binary Search Tree.

8. What is the difference between a Binary Search Tree and a Balanced Search Tree?

9. What is a Hash Table?

10. Explain heap property in Heap Sort.

5 Marks Questions

1. Compare and contrast the working of Bubble Sort, Selection Sort, and Insertion Sort with respect to their time complexity and stability.

2. Explain the working principle of selection sort with an example.

3. Write algorithm for Linear Search.

4. What is searching? Write algorithm for Binary Search.
5. Explain the working principle of bubble sort with an example.

10 Marks Questions

1. What do you mean by sorting technique? Write the algorithm for merge sort.
2. Explain the working principle of quick sort.

UNIT–6: GRAPH

2 Marks Questions

1. What is a directed graph?
2. What is an undirected graph?
3. What is the difference between a path and a cycle in a graph?
4. Explain the concept of a spanning tree in a graph.
5. What is a Directed Acyclic Graph (DAG)?
6. What is a minimum spanning tree (MST)?
7. What is topological sorting in a graph?
8. Define weighted graph.
9. Define a complete graph.
10. Define degree of a graph.

5 Marks Questions

1. Explain the differences between a directed and an undirected graph in terms of edges, degree, and applications.
2. Explain about graph terminologies.
3. Explain Topological Sorting with example.
4. Explain Prim’s Algorithm with example.
5. Explain Kruskal’s Algorithm with example

10 Marks Questions

1. Define Minimum spanning tree. Write its properties.Explain Kruskals's algorithm to find the minimum spanning tree from a graph with a suitable example.
2. Define a graph. Explain different graph terminologies.