

Lab Assignment - 1

Advanced Analysis and Design of Algorithms

Question1: Basics

Part1: Searching in lists

For the given list: 1, 4, 7, 7, 8, 9, 18, 27, 31, 56, 81

Write a program in c/c++/java to implement the following:

1. Search for 1, 81, 7, 3 in the list by scanning the list linearly and analyse the searching complexity.
2. Optimize the above searching algorithm to perform searching for the same elements in $O(\log(n))$ average time complexity by comparing search key to the middle of the list and recursively decreasing the size of search space to one half.

Part2: Sorting elements of a list

For the given list: 18, 56, 27, 7, 8, 4, 9, 31, 81, 1

Write a program in c/c++/java to implement the following:

1. Sort the list by selecting the smallest element in the list and placing it in the beginning, and repeating the process with the leftover elements until the entire list is sorted in increasing order. Also output the number of comparisons and iterations required for sorting.
2. Perform bubble sort on the same list and analyse the change in time complexity and number of comparisons required.
3. Implement Insertion sort algorithm to sort the above list and analyse the change in time complexity and number of comparisons required.
4. Given two sorted sub-halves of the list 7, 8, 18, 27, 56 and 1, 4, 9, 31, 81 implement the process of merging two sorted lists in linear time.
5. Use the above program to implement the complete merge-sort algorithm.

Part 3: Quick Sort

Implement the quick sort algorithm and show the best and worst-case time complexity on large number of elements in the array (Hint: make use chrono library in C++ to measure the time).

Question2: Recurrences

Implement the program to efficiently compute the Fibonacci sequence for a large number of terms. The Fibonacci sequence is defined recursively, but you want to analyse the time complexity and choose the most suitable method—Substitution, Master, or Recursion Tree—for understanding and optimizing your implementation.

Note: The Fibonacci sequence is defined as $F(n) = F(n-1) + F(n-2)$, with base cases $F(0) = 0$ and $F(1) = 1$.

- a) Implement the Fibonacci sequence computation algorithm using a recursive approach.
- b) Apply each of the Substitution, Master, and Recursion Tree methods to analyse the time complexity and efficiency of your algorithm.

Question 3: Probabilistic Analysis

You are the HR manager of a tech company and are responsible for hiring a software engineer for a critical project. The hiring process involves technical assessments, interviews, and evaluating the candidate's compatibility with the team. However, there is uncertainty in predicting a candidate's long-term success due to various factors.

Use a probabilistic analysis to assess the potential success of two final candidates, Alice and Bob.

Alice has a 70% chance of excelling (90% success rate) and a 30% chance of underperforming (60% success rate).

Bob has an 80% chance of performing well (80% success rate) and a 20% chance of exceeding expectations (95% success rate).

- a) Implement a probabilistic hiring strategy based on the given probabilities for Alice and Bob.

Question 4: Divide and conquer strategy

Find minimum and maximum from the array with divide and conquer strategy

Question 5: String Matching

Implement the string-matching program using

- a) Naïve algorithm
- b) Finite automata