Semester VI (B.Tech)

Jaypee University of Engineering and Technology, Guna

Advanced Programming Lab-3 18B17CI673

Lab Exercise 8: Divide and Conquer Algorithms

Title: Divide and Conquer Algorithms: An Exploration of Different Domains

Mode: Self Learning

Outcomes:

1. To explore the divide and conquer paradigm and its applications in various domains
2. To implement and compare different divide and conquer algorithms from different domains
3. To evaluate the performance of the algorithms based on their running time and memory usage
4. To gain hands-on experience in algorithm design and analysis

Methodology:

1. Selection of algorithms from different domains: Maximum Subarray Problem, Closest Pair Problem, Merge Sort, Quick Sort, Karatsuba Multiplication, and Fast Fourier Transform
2. Selection of programming language and libraries: Python and NumPy
3. Generation of random inputs with different sizes and distributions using NumPy
4. Implementation of algorithms using the selected programming language and libraries
5. Running the implemented algorithms on the selected datasets
6. Measuring the running time and memory usage of the algorithms using appropriate tools
7. Analysis of the results and comparison of the algorithms based on their performance

Steps:

1. Selection of algorithms from different domains: Maximum Subarray Problem, Closest Pair Problem, Merge Sort, Quick Sort, Karatsuba Multiplication, and Fast Fourier Transform
2. Selection of programming language and libraries: Python and NumPy
3. Generation of random inputs with different sizes and distributions using NumPy
4. Implementation of the selected algorithms using Python and NumPy
5. Running the implemented algorithms on the generated inputs
6. Measuring the running time and memory usage of the algorithms using appropriate tools such as the Python time module and memory_profiler library
7. Analysis of the results and comparison of the algorithms based on their performance using appropriate statistical analysis techniques

8. Writing a report summarizing the methodology, results, and conclusions of the experiment.

Experiments:

1. Maximum Subarray Problem: generate random arrays, implement Naive algorithm, implement Divide and Conquer algorithm, run algorithms on generated arrays, measure running time and memory usage, analyze results.
2. Closest Pair Problem: generate random points, implement Naive algorithm, implement Divide and Conquer algorithm, run algorithms on generated points, measure running time and memory usage, analyze results.
3. Merge Sort: generate random arrays, implement Merge Sort algorithm, run algorithm on generated arrays, measure running time and memory usage, analyze results.
4. Quick Sort: generate random arrays, implement Quick Sort algorithm, run algorithm on generated arrays, measure running time and memory usage, analyze results.
5. Karatsuba Multiplication: generate random integers, implement Naive algorithm, implement Divide and Conquer algorithm, run algorithms on generated integers, measure running time and memory usage, analyze results.
6. Fast Fourier Transform: generate random signals, implement Naive algorithm, implement Divide and Conquer algorithm, run algorithms on generated signals, measure running time and memory usage, analyze results.