|  |  |
| --- | --- |
| E:\SEECS\Office Work\SEECS logo\01.jpg | National University of Sciences & Technology (NUST)  School of Natural Sciences (SNS)  Department of Mathematics |

**Assignment 03**

**Project Proposal**

**CS250: Data Structures and Algorithm (3+1)**

**BS (Mathematics) : Fall 2023**

|  |
| --- |
| **Course Learning Outcomes (CLOs)**  **CLO-1:** Understand the fundamentals of data structures and algorithms  **CLO-2: Apply Data Structures and Algorithms to solve complex engineering problems.**  **CLO-3: Use appropriate Data Structures and Algorithms to design solutions**  **CLO-4:** Investigate and evaluate various algorithms based on accuracy, time complexity, and memory requirements. |

**In this assignment, you need to provide your project proposal along with group member details with task distribution. Use the template given to provide all information.**

Output:

Task – 1

Provide a 1 - page proposal of your selected project, explaining the overall aims and achievable targets. You should concisely explain the complex engineering problem you are trying to solve.

|  |
| --- |
| Implementing Fast Fourier Transform (FFT) Algorithm and Image ProcessingIntroduction:  * Brief overview of FFT and its significance in signal processing and image analysis. * Mention the goal: Implementing FFT from scratch for image processing in Python.  Objectives:  * Develop a deep understanding of the FFT algorithm and its applications. * Implement the FFT algorithm in Python. * Apply the FFT algorithm to process and manipulate images.  Proposed Methodology:  * Research and Understanding   + Comprehensive study of the FFT algorithm and its mathematical principles.   + Understanding the mathematical basis of how FFT operates on images.   + Exploring existing libraries or implementations for reference (NumPy, SciPy, etc.). * FFT Algorithm Implementation   + Step-by-step breakdown of the FFT algorithm:     - Transforming 1D arrays and understanding the iterative steps.     - Extending the algorithm to 2D arrays for image processing.   + Writing Python functions/classes for the FFT algorithm. * Image Processing with FFT   + Preprocessing images for FFT analysis (grayscale conversion, resizing, etc.).   + Applying the implemented FFT algorithm to images:     - Transforming images to the frequency domain.     - Performing operations like filtering, noise reduction, etc., in the frequency domain.     - Reconstructing images from the modified frequency domain.  Expected Outcome: Successfully implementing the FFT algorithm in Python for image processing.  Demonstrating the effectiveness of FFT in various image manipulation tasks.  A clear understanding of the trade-offs and limitations of the self-implemented FFT. |

Task – 2

Provide group member details along with task distribution, i.e. consider your group as a team undertaking this project for timely delivery to a client. You should clearly allocate primary roles to each group member.

|  |  |
| --- | --- |
| Group Member Name | Role / Tasks Allocated |
|  |  |
|  |  |
|  |  |
|  |  |

Task – 3

In this section, you need to elaborate your project idea by mapping the below mentioned concepts to your project. Please mention briefly how a particular concept will be applied in your project. This mapping may change in next few weeks as you build and expand your project, but at this stage it should be elaborate enough to quantitatively monitor your project progress on weekly basis.

|  |  |
| --- | --- |
| Topic | Application in your project |
| Data structures and algorithms |  |
| Array / linked lists |  |
| Singly / Doubly / Circular linked list |  |
| Running time complexity, function growth |  |
| Stacks and queues |  |
| Algorithm Analysis |  |
| Sorting algorithms & Recursion |  |
| Trees & Binary search tree operations |  |
| AVL trees, priority queues |  |
| Binary heaps, hash tables |  |
| Graphs and search operations |  |
| Topological sort, spanning trees |  |
| Shortest paths, Greedy algorithms |  |