**Data Structures and Algorithms**

**Project Evaluation Sheet**

Name: Veeresh S Hiremath SRN:01FE22BCS296 Roll: 440

Implementation Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Algorithm/Data Structure** | **Used? (Yes/No)** | **How and where?** | **Space Efficiency** | **Time Efficiency** |
| Arrays | Yes | Used in most of the parts of the code like for example distance array for the dijkstras algorithm, Parent array for the prime algorithm, Path identifier array, array of BST etc(In detail mentioned in the functional analysis). | 0(n) | 0(1) |
| Structures | Yes | Structure is used for storing the heterogenous data and to form our own data structures. It is used in most part of the code as vertex, node, nodeH etc. | 0(n) | 0(1) |
| Recursion | Yes | For the In-order traversal of the BST. | 0(1) | 0(n) |
| List | Yes | The implementation of the list is made use in the hash for as the collision resolution technique. | 0(n) | 0(n) |
| Stack |  |  |  |  |
| Queue |  |  |  |  |
| Binary Tree |  |  |  |  |
| Binary Search Tree | Yes | Used for the storing the patient appointment details, removing the appointments etc. | 0(n) | 0(log n) for one element.  0(n log n) for n elements. |
| AVL Tree |  |  |  |  |
| 2-3 Tree |  |  |  |  |
| Red-Black Tree |  |  |  |  |
| Trie |  |  |  |  |
| Heap | Yes | Used for Dijkstras and Prims algorithm for efficiently get the min wt vertex. | 0(n) | 0(log n). |
| Lookup Table |  |  |  |  |
| Sparse Table |  |  |  |  |
| Fenwick Tree |  |  |  |  |
| Segment Tree |  |  |  |  |
| Skip List |  |  |  |  |
| Union-Find |  |  |  |  |
| Hashing | Yes | For storing the values from the file to the main memory. Used in hospitals, hotels, and for the places in order to get them through search efficiently. | 0(n) | 0(1) |
| DFS |  |  |  |  |
| BFS |  |  |  |  |
| In-order traversal Sort | Yes | We want to sort the appointments based on the timing. | 0(1) | 0(n) |
| Bubble Sort |  |  |  |  |
| Selection Sort |  |  |  |  |
| Insertion Sort |  |  |  |  |
| Quick Sort |  |  |  |  |
| Heap Sort | Yes | We wanted to sort the appointments based on the ranking. | 0(n) | 0(n log n). |
| Merge Sort |  |  |  |  |
| Brute Force String Search |  |  |  |  |
| Rabin Karp |  |  |  |  |
| Dijkstra | Yes | Used to find the nearest Hospital. | 0(V+E) | 0(E log v). |
| Kruskal |  |  |  |  |
| Floyd |  |  |  |  |
| Warshall |  |  |  |  |
| Prim | Yes | Used to find the shortest path connecting all the hospitals. | 0(V+E) excluding the parent array including that it is 0(n2) as it is 2d array. | 0(E log v). |
| Bellman-Ford |  |  |  |  |

Number of Functions: 18

Design Techniques and Principles used: Greedy method, Transform and Conquer, Back tracking, Recursion.