

**Session 2023-2027**

**Submitted by:**

Salman Naseem 2023-CS-78

Huzaifa Arshad 2023-CS-86

**Supervised By:**

Ms. Nazeef ul haq

**Course:**

CSC200-Data Structures and Algorithms

Department of Computer Science

**University of Engineering and Technology**

**Project Title:**

Multi-Level Smart City Simulation

**Project Overview:**

The Multi-Level Smart City Simulation is a software project aimed at creating a simulation of a futuristic smart city that integrates multiple essential components, such as transportation, utility management, citizen services, emergency response, and city planning. Each module is designed using advanced data structures and algorithms to model real-world challenges effectively and ensure system efficiency.

This project will demonstrate the practical application of stacks, queues, linked lists,graphs**,** and trees while incorporating sophisticated algorithms like shortest path finding, scheduling, and resource optimization.

**Objectives**

1. Model a dynamic smart city with interconnected systems.
2. Implement efficient data structures to solve real-world challenges.
3. Showcase the integration of different modules, each leveraging specific datastructures and algorithms.
4. Create a scalable and user-friendly simulation that responds dynamically to input and changes in the city environment.

**Modules and Their Functionalities**

**1. Transportation System**

* **Purpose**: Simulate the city’s transportation network.
* **Data Structures**:
  + **Graph**: Represent the city’s roads and intersections.
  + **Queue**: Simulate traffic signals and vehicle queues.
  + **Stack**: Support undo/redo operations for road planning changes.
* **Algorithms**:
  + **Dijkstra's Algorithm / A**\*: Find the shortest path between two points in the city.
  + Traffic optimization algorithms for dynamic traffic routing.
* **Key Features**:
  + Simulate real-time traffic.
  + Handle changes in the city map with undo/redo capabilities.
  + Optimize transportation using shortest path algorithms.

**2. Utility Management**

* **Purpose**: Monitor and manage resources like electricity and water supply.
* **Data Structures**:
  + **Linked List**: Store and update user utility records.
  + **Tree (e.g., AVL/Red-Black)**: Represent the hierarchical distribution of resources (e.g., an electricity grid or water supply network).
* **Algorithms**:
  + **Dynamic Programming**: Optimize resource allocation and minimize wastage.
* **Key Features**:
  + Track real-time consumption and manage utility records.
  + Adjust resource distribution based on demand.

**3. Citizen Services**

* **Purpose**: Manage citizen information and requests.
* **Data Structures**:
  + **Hash Table**: Store citizen details for quick access (ID as the key).
  + **Queue**: Handle service requests (e.g., healthcare, complaints).
* **Algorithms**:
  + **Search Algorithms**: Locate citizen information quickly.
  + **Scheduling Algorithms**: Prioritize and allocate resources for citizen services.
* **Key Features**:
  + Maintain citizen information.
  + Efficiently process and resolve service requests.

**4. Emergency Management**

* **Purpose**: Manage emergency services for incidents like accidents, fires, or medical emergencies.
* **Data Structures**:
  + **Graph**: Represent the network of emergency service locations (e.g., hospitals, fire stations).
  + **Priority Queue (Heap)**: Dispatch services based on severity levels.
* **Algorithms**:
  + **BFS/DFS**: Find the nearest emergency response unit.
* **Key Features**:
  + Dynamic resource allocation based on emergencies.
  + Real-time simulation of emergency responses.

**5. City Development Planning**

* **Purpose**: Plan city zoning and development.
* **Data Structures**:
  + **Stack**: Support undo/redo for zoning changes.
  + **Tree (AVL/Red-Black)**: Maintain a balanced database of city zones.
* **Algorithms**:
  + **Greedy Algorithms**: Optimize land utilization.
  + **Divide and Conquer**: Handle large-scale zoning efficiently.
* **Key Features**:
  + Manage zoning and development.
  + Revert and redo actions to refine city plans.

**Integration**

* Modules communicate via a centralized City class that ensures smooth interaction between transportation, utilities, citizen services, emergency management, and planning.
* Shared data structures (like graphs for roads and emergency services) enable interconnection between systems.