Semester Project Title:

Heat Diffusion & Thermal Propagation Simulator

Student Details

Student Name	Student Reg #	Student Degree
Yazdan Ali Khan	2024665	FEE
Raiyaan Qureshi	2024531	FEE
Muhammad Hammad Shahid	2024389	FEE

---

1. Main Features

1. 2D Grid Heat Map
2. Multiple Heat Sources Support
3. Adjustable Thermal Conductivity
4. Real-time Simulation & Visualization
5. Data Logging & Temperature Graphs

6.   User roles: Admin and Viewer with distinct access privileges.

---

2. Types of Users & User-Specific Requirements

1. Viewer / User

1.1. Will be able to view the heat diffusion grid.
 1.2. Will be able to observe heat flow in real time.
 1.3. Will be able to view hottest and coldest points in the grid.
 1.4. Will be able to view sorted temperature lists.
 1.5. Will be able to load previously saved simulations.

2. Admin

2.1. Will be able to add new heat sources at any grid location.
 2.2. Will be able to remove or modify existing heat sources.
 2.3. Will be able to select a simulation method (BFS / DFS / Recursive).
 2.5. Will be able to run sorting algorithms on temperature data.
 2.6. Will be able to save simulation states into files.
 2.7. Will be able to load saved simulation data.

---

3. Requirements Breakdown

Feature 1: Heat Source Management

1.1. Ask admin for coordinates of new heat source.
 1.2. Store heat sources inside a Linked List dynamically.
 1.3. Validate that coordinates are inside grid bounds.
 1.4. Allow deletion or modification of heat source.
 1.5. Update grid after any heat source change.

Feature 2: Running the Heat Diffusion Algorithm

2.1. Represent 2D simulation area using an Array.
2.2. Convert grid into a graph using adjacency lists.
2.3. Use Queue to perform BFS heat spreading.
2.4. Use Stack to perform DFS heat spreading.
2.5. Use Recursion for natural propagation modeling.
2.6. Use Hash Map for visited-cell lookups.
2.7. Recalculate grid temperatures at each iteration.

Feature 3: Heat Data Sorting & Ranking

3.1. Extract temperature data into an Array/List.
3.2. Sort temperatures using Bubble Sort.
3.3. Sort temperatures using Insertion Sort.
3.4. Sort temperatures using Merge Sort.
3.5. Sort temperatures using Quick Sort.
3.6. Display sorted output to user.

Feature 4: Temperature Storage Using Trees

4.1. Insert temperatures into a Binary Search Tree.
4.2. Allow traversal using In-order, Pre-order, Post-order.
4.3. Provide O(log n) searching for specific temperatures.
4.4. Provide tree-based ranking.

Feature 5: Saving & Loading Data

5.1. Save full simulation output to a text file.
5.2. Save heat sources and grid state.
5.3. Read data back from file.
5.4. Reconstruct grid and heat sources on load.

---

4. Features to Coding Matrix

Sr#	Feature Name	DSA Concept Used	Operation Performed	Complexity Analysis	Variables / Objects	Functions	Lines of code
1	Heat Source Management	Linked List	Insert/Delete/Traverse				

2	Heat Diffusion Algorithm	Arrays, Graphs, BFS, DFS, Recursion, Hash Maps	Traversal + Propagation
3	Temperature Sorting	Bubble, Insertion, Merge, Quick Sort	Sorting
4	Tree-Based Storage	BST / AVL Tree	Insert + Search + Traversal
5	File Handling	File I/O	Save + Load

5. Project Screenshots

(pictures are a WIP)