ASSIGNMENT 12.1

Algorithms with AI Assistance – Sorting, Searching, and Optimizing Algorithms

AmruthSagar Vemuganti

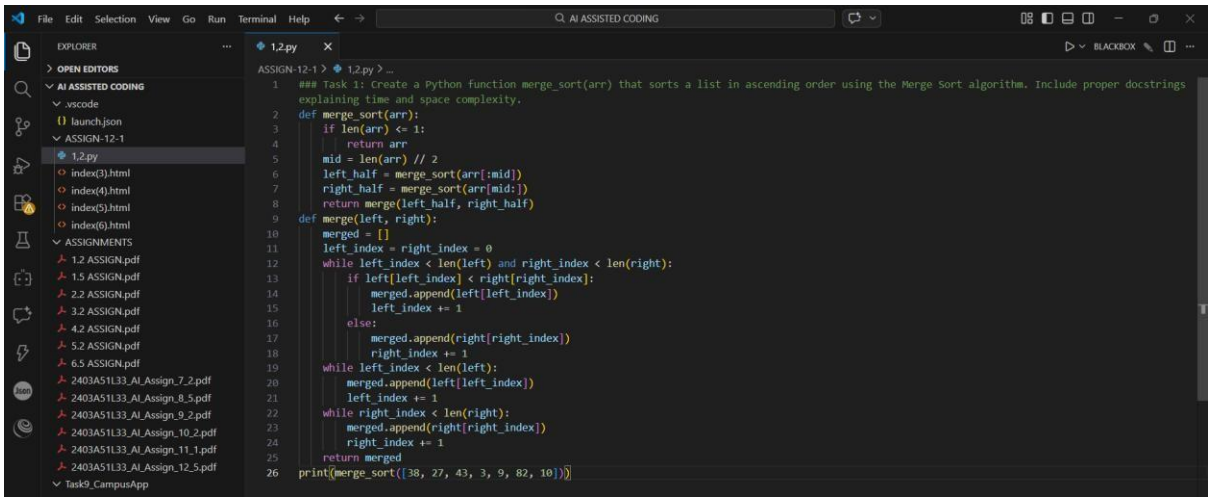
2403a51l44

B-52

Task Description #1 (Sorting – Merge Sort Implementation):

PROMPT: Create a Python function merge_sort(arr) that sorts a list in ascending order using the Merge Sort algorithm. Include proper docstrings explaining time and space complexity.

CODE:



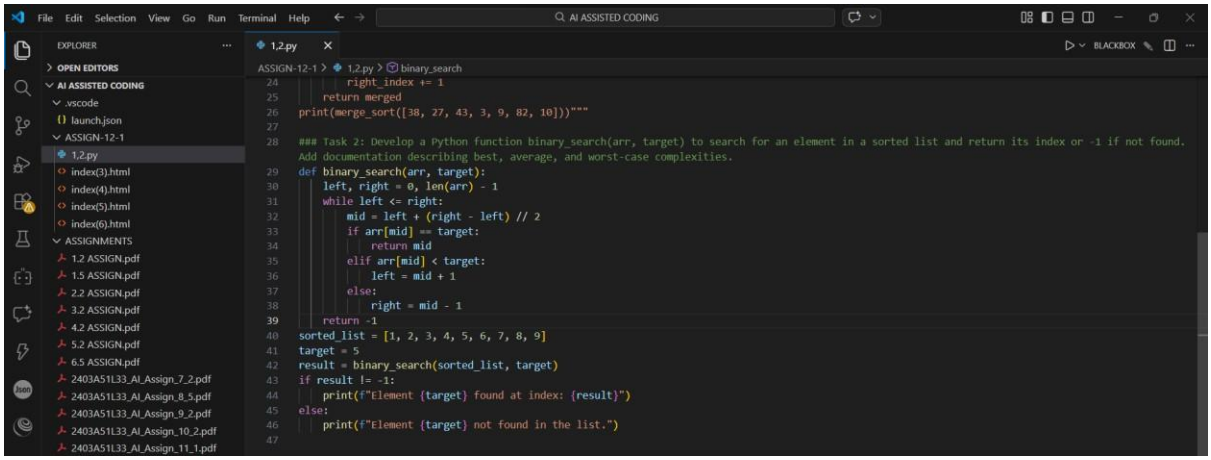
OUTPUT:

EXPLANATION: AI generated a divide-and-conquer based sorting solution with detailed complexity analysis. The output was verified using multiple test cases to ensure correctness and efficiency.

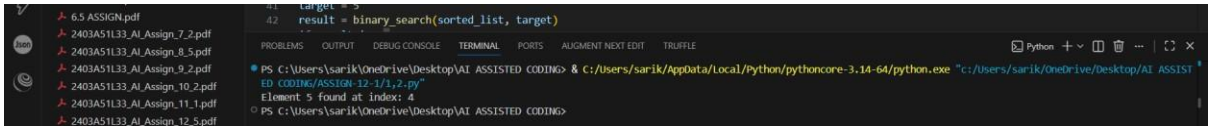
Task Description #2 (Searching – Binary Search with AI Optimization):

PROMPT: Develop a Python function binary_search(arr, target) to search for an element in a sorted list and return its index or -1 if not found. Add documentation describing best, average, and worst-case complexities.

CODE:

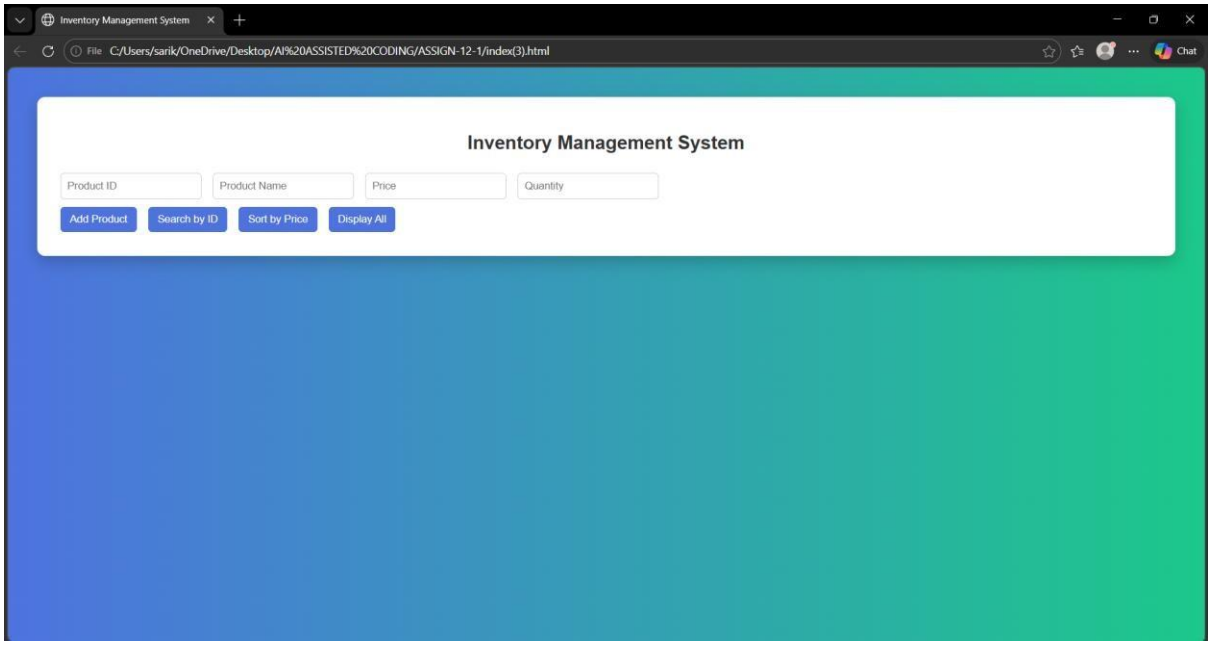


OUTPUT:

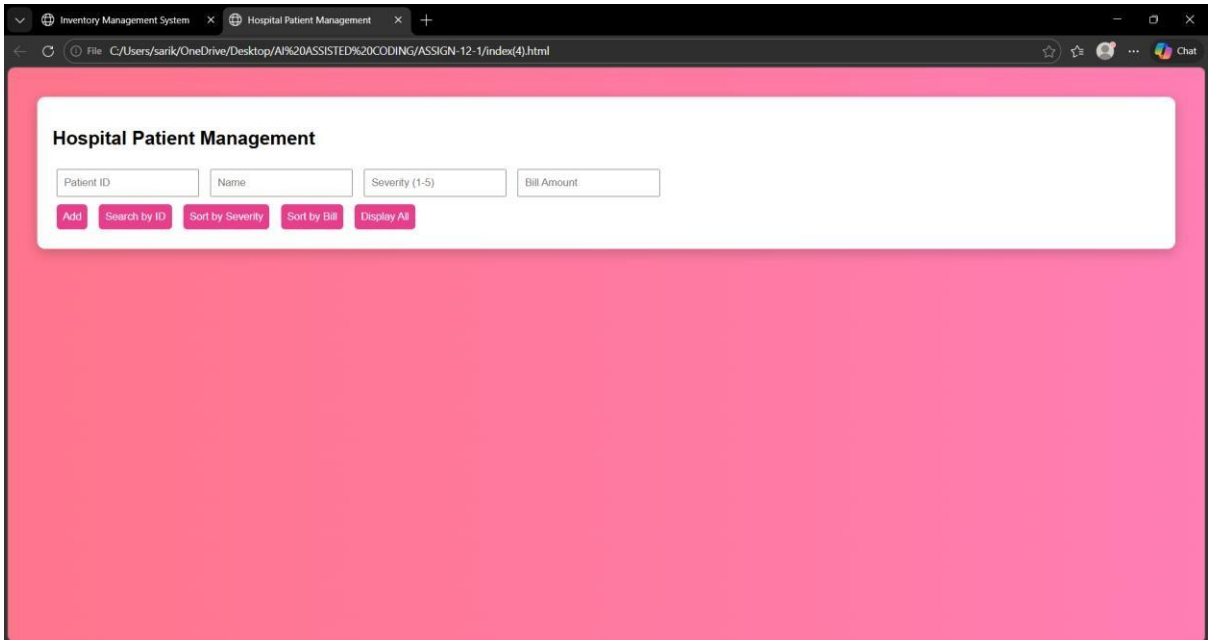


EXPLANATION: AI produced an optimized searching algorithm that reduces comparisons using a halving strategy. The function was tested with different inputs to validate accuracy and performance.

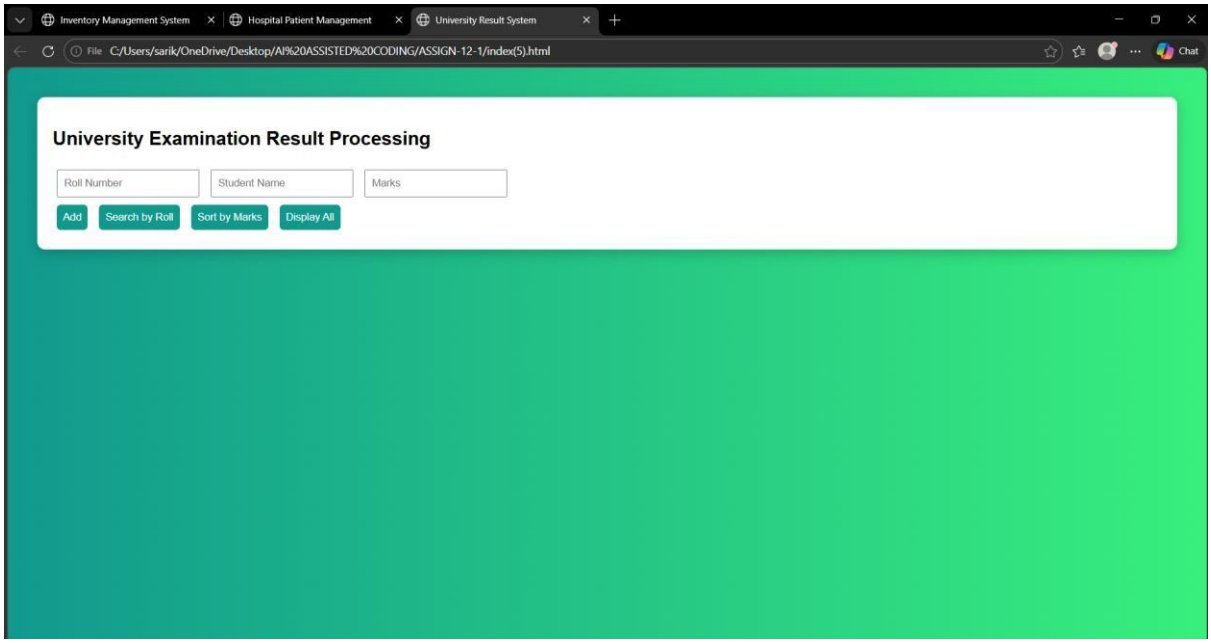
Task Description #3 (Real-Time Application – Inventory Management System):



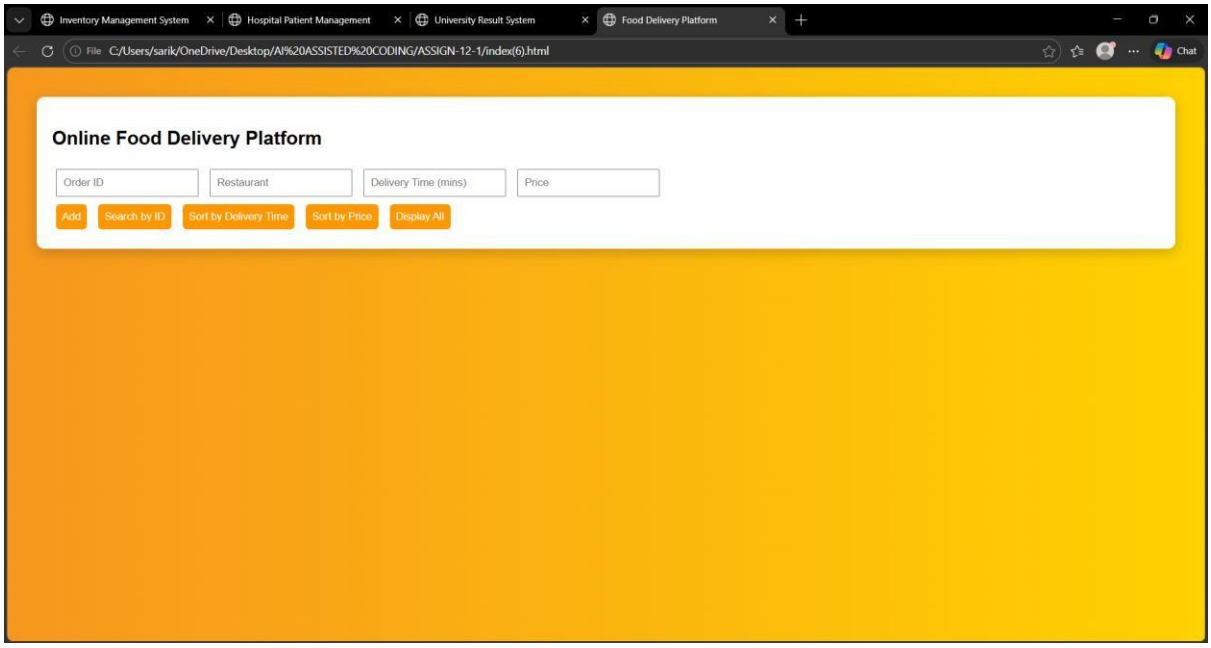
Task description #4 (Smart Hospital Patient Management System):



Task Description #5 (University Examination Result Processing System):



Task Description #6 (Online Food Delivery Platform):


Inventory Management System
Hospital Patient Management
University Result System
Food Delivery Platform
File C:/Users/sarik/OneDrive/Desktop/AI%20ASSISTED%20CODING/ASSIGN-12-1/index(6).html
Chat
Online Food Delivery Platform
Order ID
Restaurant
Delivery Time (mins)
Price
Add
Search by ID
Sort by Delivery Time
Sort by Price
Display All