Ahmed Tarek Mohamed *Machine Learning*

✉ ahmef54a@gmail.com

☎ 01012272487

Dakhliya, Mit Gamer

Egyption

delay

linkedin.com/in/ahmed-tarek-03218631a

https://youtube.com/@ahmed-r8x6u?
si=VY4UgYbS0RqcSNTZ

Education

2024 – present **GPA: 3.50**
Egyptian Chinese University

Skills

Programming Languages
C++ , Python , SQL
Concepts
Data Structure, OOP, Algorithms ,Data Base,
Computer Architecture, Operating System, Logic

Languages

- Arabic (Native)
  English (Fluent)

Projects

Two Stack In One Array
Data Structure
- Implemented two independent stack data structures within a single array to optimize space utilization.
- Developed dynamic push and pop operations for both stacks while ensuring they do not overlap or cause overflow.
- Applied pointer manipulation and boundary checks to handle memory efficiently without using extra space.
- Gained strong understanding of array-based memory management and stack operation

Snake Game with AI Pathfinding (BFS)
AI

- Developed an advanced Snake game that uses **Breadth-First Search (BFS)** algorithm to automatically guide the snake towards food while avoiding obstacles and self-collision.
- Implemented **AI-based pathfinding logic** for dynamic direction updates based on shortest path calculation.
- Added interactive **GUI for difficulty selection** and **game over/restart menus** using Tkinter.
- Designed game mechanics with randomized food, obstacles, and snake growth, including real-time score and high-score tracking.
- Gained practical experience in algorithms, game development, AI decision making, and Python graphics

Solar System Simulation with OpenGL
Computer Graphics
- Developed a 3D interactive simulation of the Solar System using OpenGL in C++.
- Modeled planets, moon, and sun with realistic textures and orbital mechanics.
- Implemented camera controls, zoom, and planet focus switching for a fully immersive experience.
- Added advanced graphical features including Saturn's rings, day-night shading (terminator effect), and starfield background.
- Used SOIL library for texture loading and GLU quadrics for smooth sphere rendering.

- Enabled real-time simulation speed control and automatic camera flythrough.

CPU Scheduling Algorithms Simulator

Operating System

- Designed and implemented a comprehensive CPU scheduling simulator in C++ covering: □ First Come First Serve (FCFS) □ Shortest Remaining Time First (SRTF) □ Round Robin (RR) □ Non-Preemptive Highest Priority First (HPF)
- Generated dynamic process data using **normal and Poisson distributions** to simulate realistic arrival/burst/priority scenarios.
- Calculated and displayed **waiting time** and **turnaround time** for each algorithm.
- Integrated file I/O to read process parameters and export simulation results.
- Strengthened understanding of scheduling behavior, time management, and algorithm performance analysis