M. Hamza Noor

+92(321) 750-0752
hamzarandhawa553@gmail.com

Website

LinkedIn GitHub
LeetCode

An Engineer who loves working on computational Algorithms and engineering applications with features driven by Generational AI models. Driven by the enthusiasm to learn more and become part of revolutionizing AI for the true betterment of lives on the planet.

EXPERIENCE

FRAG Games, Lahore— Game and AI Algorithms Engineer

July 2022 - Present

- Optimized AI algorithms like Path-finding for cross-platform AA games in Unity C# by developing the Parallel computing algorithm Jump Point Search (JPS) which proved to be two magnitudes faster than previously used A*. Details are mentioned below.
- Developed the Boids AI algorithm for the flocking behavior of NPCs and troops in games.
- Developed Base Building of a AAA turn-based tactical strategy role-playing game.
- Wrote Unit Tests for an AA game using Unity Unit Testing Framework implementing the TDD practice.
- Worked on the Porting of an AA game from PC to cross platforms.
- Worked on the role-playing adventure game WrestleQues for the developer Mega Cat Studios and publisher Skybound Games.
- Optimized GPU rendering and Physics engine rendering in MagaCat's Wrestle Quest game.

Black Hawkes Studio, Lahore— Unity Game Developer

January 2022 - June 2022

Developed Mobile Games for iOS and Android. Solely Built Dangerous Roads details of which are mentioned below.

DeltaLabs, Lahore— Ruby on *Rails Developer*

January 2021 - January 2022

Managed Back and Front end of reactive web apps with complex database structures using Ruby on Rails and React.

EDUCATION

BACHELOR IN COMPUTER SCIENCE

University Of Central Punjab, Lahore | OCT 2016 - JUL 2020

- CGPA: 3.53
- Included in Dean's Honors List
- Secured A grade in but not limited to; Programming Fundamentals, AI,

TECHNICAL SKILLS

Programming Languages:

- C#
- C++
- SQL
- R
- Java
- JavaScript
- Assembly
- Ruby
- c

Game Development

- Unity Game Engine
- Unreal Game Engine
- Inkl

Tech Stacks

- PyTorch
- NodeJS
- React
- Ruby on Rails
- MySQL

Android Apps Development

- Android Studio

IoT Development

- Arduino Platform

Data Structures, Algorithms, Object-Oriented Analysis, Database, and Computer Organization and Assembly Language.

PUBLICATIONS

H. Noor, “Node Pruning by Jump Points Optimize A* by One Magnitude and More”, Manuscript under Review, September 2024

PROJECTS

ChattyVR— 2023-Present

Under the supervision of Dr. Aamer Zaheer—a Computer Vision Architect at Apple, I engineered this exciting application in the Unity engine. In this application, a user verbally chats with a talking person whose conversation is driven by openAI’s GPT-4 model using API. Making the chatbot person’s gestures coherent with the voice and minimizing the time of voice response are challenges that I am still working on.

Jump Point Search Algorithm— 2022-2023

In pursuit of solving the latency problem in finding paths in AA games complex 2D grid maps in FRAG Games, engineered a new algorithm optimizing the A* algorithm by carefully only exploring those nodes which could be turning or jump points.

On the grid of 10,000 * 10,000, Jump Point Search (JPS) is two magnitudes faster than A*. And it is solving big real-time latency problems in many projects.

WrestleQuest— 2023

WrestleQuest is a role-playing video game developed by Mega Cat Studios and published by Skybound Games. It combines a pro-wrestling theme with elements from Japanese role-playing games. I worked on it in its release phase to solve its bugs. I resolved performance related issues by optimizing GPU rendering and Physics engine rendering.

Boids— 2022

Implemented a life program simulating the behavior of a flock of birds, school of fishes, or herd of land animals designed by Craig Reynolds in his paper Flocks, Herds, and Schools: A Distributed Behavioral Model (1986). The Program adds behaviors of Separation, Alignment, and Cohesion in NPCs.

Dangerous Roads— 2022

An arcade game in which you have to dodge traffic, collect logs, and make bridges to help pedestrians pass the dangerous roads. Passing pedestrians will give you rewards with which you can update bridges and unlock new areas

Video Link

CricInsights — 2020

All-time Cricket Players Comparison Application built in R Studio. The user picks any two players, batsmen, or bowlers of any time, their data from the database gets fetched and statistical comparison is shown to the user in the table and the graphs are shown to graphically visualize that.

GitHub Link
Published Application Link

Friend Snakes — 2018

Android application of Double Player Snakes Game with one player controlled by a human and the other controlled by the program with Artificial Intelligence implemented using BFS(Breadth First Search) method. Developed on Android Studio.

Desktop App GitHub Link

Android App GitHub Link

Path Finding Car — 2020

Pathfinding (Object avoidance) car with infrared sensors with Arduino chip. It is built in the Arduino platform with Programming in C language.

Remote Control Car — 2020

Arduino-based project of a car controlled by Remote and with additional infrared sensor to avoid hitting objects.

Minesweeper Game— 2018

Desktop Application of Mine Sweeper Game, Human Vs Computer in C# with DIfficulty Options Easy, Medium, and Hard.

GitHub Link

Tic Tac Toe Game — 2016

With User decision, Human vs Human double player game or Human Vs Computer (with Artificial Intelligence) double player game of Tic Tac Toe with NxN board. N will be chosen by the user. It has Game Save & Load features using and handling the file system. It is a Console application in C++.

GitHub Link

Chess Game — 2017

Double Player Chess Game with valid moves suggestions and Save & Load Game and board features using and handling File system. It is a Console application in C++.

GitHub Link

Algorithms— 2019

- **Recursive Algorithms with Memoization Technique (Dynamic Programming):**

1. Fibonacci with Memoization (Dynamic Programming)
2. Tri Series Sum with Memoization (Dynamic Programming)
3. N- Series Sum
4. Permutation (nPr) and Combination (nCr)
5. Linear Search
6. Binary Search
7. Fast Powering
8. Multiplication by Addition

GitHub Link

- **Recursive Algorithms with Divide and Conquer Technique:**

1. Merge Sort
2. Merge Sort For Linked List
3. Find a Value in a Matrix
4. Determinant of a Matrix

GitHub Link

- **Karatsuba Algorithm:**

  Fast Multiplication algorithm for very long digit numbers.

GitHub Link

- **Optimized Algorithms:**

1. Recursive Tower of Hanoi
2. Ai+Aj =x
3. Sorting with K different Values
4. Finding a Reversing Circular Shifting
5. Contiguous Sub-Array with Maximum Sum
6. Index at which V[i] = i
7. Which X[i] and Y[j] values can be swapped to equate X and Y

GitHub Link

- **Prison Break Problem:**

  Prison Break problem solved with shortest path using 2D Vectors and STL Queues.

GitHub Link

Red Black Tree — 2018

Self-balancing binary search tree made Data Structure and Algorithms course. Each node stores an extra bit representing color, used to ensure that the tree remains approximately balanced during insertions and deletions. Searching, Insertion and Deletion take place in O(log n) and Space is in O(n).

GitHub Link

Snake and Ladders Game — 2017

With User decision, Human vs Human double player game or Human Vs Computer double player game of Snake and Ladders with board 1 - 100 board. It has Game Save & Load features using and handling the file system. It is a Console application in C++.

GitHub Link

Ping Pong Game — 2017

Console application in C++ of Two Player Ping Pong game with Save and Load functions using and handling File system.

GitHub Link

Green Force VR (FYP)— 2019

A free-world VR Game for the impact to create awareness against the destruction of the environment and pollution in today's world. A near-future era, the year 2034 is shown to depict the bad consequences of polluting human activities. Our player avatar is a member of an emergency task force named Green Force. Green Force is a team of Jetmen with powered jetsuits. The task of the Green Force is to handle emergency situations of Pollution and the Environment. When the player of the Game solves pollution problems in the game, then she or he aso starts thinking about the same problems and their solutions in real life.

MUX and DEMUX— 2017

16x4 Multiplexer (Data Selector selecting between 16 digital input signals and forwards the selected input to a single output line) and 4x16 Demultiplexer (Data Distributor circuit that receives data on a single line and transmits this data on multiple output lines; binary of the Input) built on hardware.