Kasra Kashani

benkasra@gmail.com | (+98) 919 199 1418

Linkedin | Github

Education

University of Tehran, BS in Computer Engineering   Sept 2022 – Now
- GPA: 18.77/20 (3.75/4)

Salam Tajrish Highschool, Diploma in Mathematics and Physics   Sept 2019 – Sept 2022
- GPA: 19.61/20 (3.92/4)

Research Interests

Artificial Intelligence	Machine Learning	Automata and Language Theory
NLP	Data Science	Compiler Designing

Honors and Awards

- **TOP 10% among 110 Computer Engineering B.Sc. students**
  Member of Elite Students, University of Tehran

- **TOP 100 among 145,000 participants in the National University Entrance Exam**
  Known as Konkur of Mathematics and Physics, Sanjesh Org

Relevant Courses

Course	Term
Data Science, *Instructor: Dr. Bahrak & Dr. Yaghoobzadeh*	In Progress
Compiler Designing and Programming Languages, *Instructor: Dr. Tavassoli*	In Progress
Artificial Intelligence, *Grade: 19/20, Instructor: Dr. Fadayi & Dr. Yaghoobzadeh*	Fall 2024
Computer Aided Design, *Grade: 20/20, Instructor: Dr. Modarresi*	Fall 2024
Algorithm Design, *Grade: 19.4/20, Instructor: Dr. Asadpoor*	Spring 2024
Formal Language and Automata Theory, *Grade: 18.2/20, Instructor: Dr. Hojjat*	Spring 2024
Computer Architecture, *Grade: 19.5/20, Instructor: Dr. Safari*	Spring 2024
Data Structures and Algorithms, *Grade: 20/20, Instructor: Dr. Faili & Dr. Amiri*	Fall 2023
Engineering Probability and Statistics, *Grade: 19.8/20, Instructor: Dr. Tavassolipour*	Fall 2023
Advanced Programming, *Grade: 18.2/20, Instructor: Dr. Khosravi*	Spring 2023
Introduction to Computing Systems and Programming, *Grade: 20/20, Instructor: Dr. Hashemi*	Fall 2022

Teaching Experiences

Computer Architecture   Fall 2024 – Spring 2025
Instructor: Dr. Saeed Safari   *University of Tehran*
  - Teaching Assistant: Homework, quiz and computer assignment grader

Formal Language and Automata Theory   Fall 2024 – Spring 2025
Instructor: Dr. Hosein Hojjat   *University of Tehran*
  - Teaching Assistant: Homework designer and grader

Introduction to Computing Systems and Programming   Fall 2024
Instructor: Dr. Mahmoud Reza Hashemi   *University of Tehran*
  - Teaching Assistant: Laboratory TA, Final exam grader

Skills

Programming

  - **Advanced:** Python, C++, C, Java

  - **Intermediate:** VerilogHDL, SystemVerilog, LaTeX

  - **Beginner:** SQL, Assembly, HTML, CSS

Tools

  - Quartus, Multisim, Modelsim, Vivado, Linux, Git, FPGA boards, Jupyter Notebook, Tableau, PowerBI, Kafka, Apache Spark, Docker, Kubernetes, Spring, MySQL, MongoDB, Star UML, Matlab, Word, PowerPoint, Excel

Libraries

  - PyTorch, TensorFlow, Scikit-Learn, Pandas, Numpy, Matplotlib, Seaborn, SciPY, PySpark, PyMongo, Selenium,

Interpersonal Skills

  - Leadership, Teamwork, Problem-solving, Social skills, Critical thinking, Time management, Fast learning

Notable Projects

Compiler Designing for a new language CPY
Implementing the frontend of a compiler for a C-Python-like language using Java. It includes a full-featured Abstract Syntax Tree (AST), grammar and parsing modules, semantic analysis via visitor pattern, and a robust symbol table implementation.
  - Language: Java

Fake News Detection
This project implements a Fake News Detection system using a Python-based data pipeline. It includes preprocessing, feature engineering, and classification to identify misinformation.
  - Language: Python

AI-Powered Connect4 Game with Minimax Algorithm
Designed an AI-driven Connect4 game using the Minimax algorithm with alpha-beta pruning and a customizable difficulty. Features a Pygame interface for an engaging user experience.
  - Language: Python

Lights Out Puzzle Heuristic Search
Analyzed various heuristic search algorithms, including BFS, IDS, A*, and Weighted A*, for solving the Lights Out Puzzle. Focused on optimizing search performance, node exploration, and execution time.
  - Language: Python

Genetic Image Reconstruction
Developed a program that recreates images using genetic algorithms with mutation, crossover, and fitness evaluation for image refinement.
- Language: Python

Building Neural Network from Scratch
Implementing the core components of deep learning pipelines, including forward and backward passes, loss functions, optimizers, and training routines, without using deep learning libraries like TensorFlow or PyTorch.
- Language: Python

CNN for Image Classification with VGG16
This project demonstrates the application of Convolutional Neural Networks (CNNs) for image classification on the CIFAR-10 dataset, using transfer learning with the VGG16 architecture. The task is to classify 60,000 32×32 color images across 10 distinct classes.
- Language: Python

Hotel Price Prediction Classification ML
This machine learning project focuses on predicting hotel price categories based on various features such as hotel type, location, star rating, facilities, and user reviews. The goal is to use classification models to group hotels into distinct price brackets (low or high) and help users or agencies understand pricing trends.
- Language: Python

News Subject Clustering
An unsupervised clustering project using sentence embeddings from a pretrained transformer model and classic clustering techniques and then group news texts based on semantic similarity, enabling automatic categorization and exploration of textual datasets.
- Language: Python

Roulette Simulation Profit Analysis & Predicting 2016 USA Election & Drug Safety Test
This project consists of three data science mini-projects focused on statistical simulation, inference, and data analysis, implemented using Python and Pandas.
- Language: Python

Sampling & Airbnb Storytelling using Tableau
This project consists of two data science mini-projects composed of two main tasks: Langevin Dynamics Sampling Algorithm, and Airbnb Data Storytelling and Visualization in Tableau.
- Language: Python

Real-time Payment Data Pipeline using Kafka
This project is a comprehensive suite of real-time streaming applications built using Apache Spark Structured Streaming and Kafka, designed to perform advanced analytics on financial transaction data.
- Language: Python

Cancer Patient Survival Prediction Classification ML
Developed a machine learning pipeline to predict cancer patient survival status using clinical, demographic, and treatment data. The goal is to classify each patient as either Alive (1) or Deceased (0).
- Language: Python

Bike Rental Prediction Regression ML
Built a supervised machine learning regression model to predict the number of daily bike rentals based on weather, seasonal, and calendar-related features.
- Language: Python

Movie Recommender System ML
Implemented a trust-aware movie recommender system using traditional machine learning techniques (non-deep learning) to predict user ratings for unseen movies.
- Language: Python

Predict 2022 Fifa World Cup using MLP DL
This project trains a Multi-Layer Perceptron (MLP) model to predict the outcomes of football matches during the road to the Qatar 2022 FIFA World Cup and the tournament itself. The model is trained on qualifiers and used to simulate the full tournament.
- Language: Python

Flower Species Image Classification using CNN DL
Focused on classifying flower species using Convolutional Neural Networks (CNNs). Two key approaches were used: training a custom VGG-style CNN from scratch, and fine-tuning a pretrained ResNet50. The results of both were compared using various performance metrics.
- Language: Python

Forecast Future Bitcoin Prices using RNN DL
Focused on time series forecasting of Bitcoin prices using a Recurrent Neural Network (RNN). The project walks through EDA, feature engineering, RNN modeling, and performance evaluation. An LSTM variant is also optionally implemented for improved performance.
- Language: Python

Hokm Game
Implementing a complete 4-player version of the traditional Persian card game Hokm. Players take turns playing cards according to official rules, and teams compete to win hands and ultimately the game.
- Language: C

UT2T Messenger
A terminal-based messaging application written in C that allows users to sign up, log in, and interact with posts through a simplified social interface. It mimics the structure of a microblogging service with essential features like posting, liking, and managing content.
- Language: C

UTfood
Implemented a simplified console-based version of a food delivery system similar to SnappFood. It includes features for user registration, login, account management, restaurant and food menu management, and discount codes. The system supports three user roles: admin, customer, and restaurant.
- Language: C++

Class Scheduler
Implemented an automated weekly class scheduling system designed to assign course times and instructors to university classes without any conflicts. It is built to manage instructor availability, classroom time slots, and course requirements effectively.
- Language: C++

Place Visit Planner
Designed to efficiently plan and optimize a schedule for visiting a set of places within a given day based on constraints such as opening/closing times and place rankings.
- Language: C++

Mini Snapp-like Transportation Missions
Simulated a simplified version of a ride dispatch system where a central controller (Snap) manages missions and coordinates with drivers, similar to a logistics platform.
- Language: C++

Football Fantasy
A simulation of a Premier League Fantasy Football game. It allows users to manage fantasy teams, simulate matchweeks, and track scores and player performances using real-world data.
- Language: C++

Harry Potter Audio Signal

This project focuses on the analysis, processing, and classification of audio signals from Harry Potter movie scenes. The objective is to use machine learning and signal processing techniques to classify audio clips based on spoken characters, background music, or scene types.

- Language: Python

Operating System Laboratory Projects
Worked with the xv6 Unix-like OS, focusing on kernel-level modifications and feature extensions. Projects included implementing and testing custom system calls, CPU scheduling, terminal I/O, memory management, and synchronization.

- Language: C

Operating System Projects
Explored socket programming and network communication for inter-process data exchange. Developed MapReduce systems and multi-threaded applications to implement distributed and parallel computing.

- Language: C++

Statistics and Probability Projects
Applied Bayes' theorem with text preprocessing and worked on distributions including binomial, Poisson, and normal. Covered Bayesian estimation, regression, and the Central Limit Theorem in analytical projects.

- Language: Python

Data Structures Projects
Built and tested data structures like stacks, queues, linked lists, heaps, trees, and graphs. Focused on recursive algorithms and time complexity analysis for efficiency.

- Language: Python

Algorithm Design Projects
Implemented algorithms using divide and conquer, dynamic programming, greedy techniques, and graph-based methods. Explored NP-complete problems and approximation algorithms.

- Language: Python

Computer Aided Design Projects
Designed an approximate multiplier and circular buffer, and synthesized modules using ACTEL. Implemented a processing array modeled after the Eyeriss accelerator. Also implementing a Maclaurin Series calculator using Pipeline.

- Language: VerilogHDL, SystemVerilog

Digital Logic Design Projects
Developed RTL modules for logic components and accelerators. Simulated and synthesized combinational and sequential circuits including gates, latches, and flip-flops.

- Language: VerilogHDL

Digital Logic Design Laboratory Projects
Built accelerators with wrappers, designed function generators, and implemented periodic signal generation. Synthesized sequential logic on FPGA platforms.

- Language: VerilogHDL, SystemVerilog

Computer Architecture Projects
Designed and simulated single-cycle, multi-cycle, and pipelined RISC-V processors. Implemented arithmetic units and integrated control and datapath for instruction execution.

- Language: VerilogHDL

Computer Architecture Laboratory Projects
In our Computer Architecture Lab course, we designed a pipelined ARM processor and implemented it on a development board. The processor includes hazard detection and forwarding units to handle data dependencies efficiently

- Language: VerilogHDL, SystemVerilog

Languages

English: Full professional proficiency
Persian: Native