Tamajit Banerjee

tamajitbuba3@gmail.com | +91-7003934884 | GitHub | LinkedIn

Academic Details

Year	Degree/Board	Institute	GPA/(%)
2019-23	B.Tech. in Computer Science & Engineering	Indian Institute of Technology, Delhi	9.67
2019	CBSE XII	South Point High School , Kolkata	94.4%
2017	CBSE X	South Point High School , Kolkata	10

Scholastic Achievements

- Awarded **IIT Delhi Semester Merit Award** for being among the **Top 7**%students in the 2019-20 Sem I and Sem II.
- Secured **Global Rank 26** in **Google Kickstart Round B** 2022 and Qualifed for **Round 3** of **Facebook Hacker Cup** 2021.
- Achieved **AIR 80** in **JEE ADVANCED 2019** and **AIR 141** in **JEE MAINS 2019** among 1.6 million candidates.
- Secured **2nd** position in **West Bengal Joint Entrance Examination** (WBJEE) 2019 among 0.125 million candidates.
- Achieved **AIR 54** in **KVPY-SX** category in 2018 and **AIR 112** in **KVPY-SA** category in 2017 among 0.1 million candidates.
- Recipient of **NTSE scholarship** from the **Govt. Of India**, for being among **Top 1000** students out of 1 million candidates.
- Secured a place in **National Top 1**%in NSEP (**Physics**), NSEC (**Chemistry**), and NSEA (**Astronomy**) **Olympiads** in 2018.

Internships

- **Google, India** | Software Engineering Internship May, 2022 - July, 2022
  Low Dependency Consistent and Dynamic Resharding Schemes | Infrastore Team, Cloud
  - Designed and developed an end-to-end cluster level **service** that assigns and reassigns the shards to balance out the load.
  - Minimised the **downtime** of service and proposed using **Maximum Weighted Bipartite Matching** for faster resharding.
- **Max Planck Institute for Software Systems, Germany** | Reseach Internship May, 2021 - July, 2021
  Symbolic Algorithms for Two-Player Games under Strong Transition Fairness | Prof. Rupak Majumdar
  - Designed and implemented an end-to-end **BDD-based synthesis engine** in **C++** using the Sylvan and CUDD libraries.
  - **Parallelised** the operations and **used intermediate computations** which improved the **run time** by **100 times**.
  - Showed that it is **scalable** by testing it on real and standard synthetic **benchmarks** and constructed code-aware resource managers for **multithreaded embedded** applications and our research paper was published at **TACAS 2022**.

Research Projects

- **Automated Surgical Skill Evaluation** | Prof. Chetan Arora | Python *November, 2021 - March, 2022*
  - Analysed SOTA for surgical skill assessment on benchmark JIGSAWS dataset, aiming to generalize to NETS dataset.
  - Used rank-aware loss to improve the learnt representation space and make loss more relevant to evaluation metric.
  - Showed improvements over baseline performance on both datasets and our research paper was published at ICIP 2022.
- **Surgical Video Instance Segmentation** | Prof. Chetan Arora | Python *March, 2022 - July, 2022*
  - Worked on video instance segmentation for surgical instruments on the benchmark EndoVis 2017 and 2018 challenge.
  - Critically analyzed around 10 methods publised in top vision conferences including MaskRCNN, MaskProp, SG-Net etc.
  - Introduced a 3rd stage specializing in classification to correct the label of prediction masks obtained from 2nd stage.
  - Improved upon all the above methods compared with, and achieved at least 17 points (30%) improvement over SOTA

Course Projects

- **Multi-Player Game and Simulation** | Prof. Rijurekha Sen | C++ *April, 2021 - May, 2021*
  - Designed and developed a two-player competitive maze game from scratch using SDL and Socket Programming.
  - Worked on generating pseudo-random numbers and randomised maze, all-pairs shortest paths and animations.
  - Explored heuristics for Cost-Constrained Travelling Salesman Problem with simulation to visualize and compare them.
- **Basic Operating System Kernel** | Prof. Sorav Bansal | C *January, 2022 - April, 2022*
  - Designed the Shell Program and added features for debugging routines by I/O mapped devices and keyboard drivers.
  - Implemented Coroutines, Fibers and Preemption enabling our shell to be responsive while performing computations.
  - Created a SPSC queue to manage shared data structures between cores and enabling multi-core programming functionality.
- **Reading (Noisy) Captions Embedding in Images** | Prof. Parag Singla | Python *October, 2021 - November, 2021*
  - Used a CNN encoder to handle variable sizes and LSTM decoder to generate the captions given the encoded image.
  - Used cross-entropy as loss function and trained decoder using teacher forcing and used beam search for extraction.
- **Object Localisation from noisy observations** | Prof. Rohan Paul | Python *February, 2022 - March, 2022*
  - Used Kalman Filter to estimate the trajectory of an object having a prior belief of its initial state and noisy observations.
  - Estimated the most-likely path and the current position of a robot in a grid by modelling it as a Hidden Markov Model.

- **DRAM request manager for MIPS simulator** | Prof. Preeti Ranjan Panda | C++ *March, 2021 - May, 2021*
  - Simulated a Multi-Core Processor that supports a subset of MIPS instructions and a non-blocking memory algorithm.
  - Optimized ordering of requests using FR-FCFS Scheduler, priority encoding, and forwarding to maximize throughput.
- **Traffic Density Estimator using OpenCV** | Prof. Rijurekha Sen | C++ *February, 2021 - March, 2021*
  - Used homography to change perspective and background subtraction and optical flow to estimate the traffic density.
  - Analysed utility vs runtime trade-off by varying FPS, frame resolution, and multi-threading (spatial and temporal).
- **Parallel Template Search in Image** | Prof. Subodh Kumar | C *March, 2022 - April, 2022*
  - Implemented a system to determine the position of potentially rotated query images in input image concurrently.
  - Employed CUDA for parallel computation, bilinear interpolation to compute coordinates, RMSD computations..
- **Template Tracking** | Prof. Chetan Arora | Python *September, 2021 - October, 2021*
  - Implemented block-matching techniques and pyramid-based Lucas Kanade algorithm using multiscale Gaussian pyramid.
  - Created a UI using Tkinter library to live track affine and projective transformation to an object using bounding box.
- **Pedestrian Detection in images** | Prof. Chetan Arora | Python *October, 2021 - November, 2021*
  - Prepared training data for pos. and neg. samples, extracted Histogram of Oriented Gradients features using scikit library.
  - Trained a linear SVM classifier, used multiscale sliding window technique, compared with pre-trained Faster-RCNN.
- **Reinforcement Learning for online policy learning** | Prof. Rohan Paul | Python *November, 2021 - December, 2021*
  - Modelled the problem as Markov Decision Problem, implemented dynamic programming to compute good policy offline.
  - Compared Q-learning and SARSA in epsilon-greedy policy with both fixed and decay epsilon to learn in unknown domain.
- **Simulating Dynamic Memory Allocation** | Prof. Rahul Garg | Java *October, 2020 - November, 2020*
  - Implemented the First Split Fit and Best Split Fit, and Defragmentation methods of dynamic memory allocation.
  - Analyzed the time complexity of the methods for using doubly-linked lists, binary search trees, and AVL-Trees.
- **Evaluation of a functional programming language** | Prof. S. Arun Kumar | SML *March , 2020 - April, 2020*
  - Built the basic system used in search engines: inverted index using doubly-linked lists, hash table, and AVL-Trees.
  - Implemented the web search problem of compound searches by returning the pages in the order of decreasing "relevance".
- **File Transfer through Serial Receiver/Transmitter** | Prof. Anshul Kumar | VHDL *April , 2022 - May, 2022*
  - Designed and implemented an asynchronous serial receiver and transmitter, FIFO Buffer and 7 Segment Display separately.
  - Combined them to build an end-to-end service for downloading (PC to BASYS-3) and uploading files (BASYS-3 to PC).
- **Digital Image Filtering** | Prof. Anshul Kumar | VHDL *December , 2020 - January, 2021*
  - Applied sharpening and smoothening operations on an image (8-bit pixel resolution) using a 3x3 sliding window in VHDL.
  - Processed the image using the coefficients from ROM and a multiplier accumulator (pipelining) and stored it in RAM.
- **A Small Search Engine** | Independent Project | Java *November, 2020 - December, 2020*
  - Implemented a toy programming language capable of lexing, parsing, evaluating, and type-checking arithmetic expressions.
  - Supports user-defined variables, scope rules, higher-order and anonymous functions, and error handling.

Key Courses Undertaken

Introduction to Computer Science **A**
Probability and Stochastic Processes **A**
Programming Languages **@**
Computer Vision **A**
Computer Networks **A-**
Intro. to Parallel and Dis. Pro. **@**
Linear Algebra and Differential Equations **A**
Discrete Mathematics **@**
Computer Architecture **A**
Artificial Intelligence **A-**
Special Topics in Artificial Intelligence **A**
Theory of Computing **A-**
Data Structures and Algorithms **A**
Digital Logic and System Design **A-**
Analysis and Design of Algorithms **A**
Machine Learning **@**
Operating System **A-**

A: Max. Grade | **@**: Audit

Technical Skills

- **Languages :** C++, C, Python, Java, Bash, Standard ML, VHDL, MIPS, x86, LaTeX, SQL, HTML.
- **Softwares/Libraries :** Git, OpenCV, Numpy, Pandas, SciPy, Scikit-Learn, TensorFlow, Pytorch, Shell Utilities, OpenMP.

Position of Responsibility

- **Coordinator, Algorithm and Coding Club** and design original and challenging problems. *March, 2021 - Present*
- **Hostel Chess Captain** and mentored a team through weekly practice sessions and evaluations. *January, 2021 - April ,2022*
- **Sports Committee Member** and helped in improving sports facility and sport resources. *January, 2021 - April, 2022*

Extra Curricular Activities

- Regular **participation** in various online coding competitions. **Codechef** Rating: **2347** | **Codeforces** Rating: **2121**.
- Provided guidance to 40+ freshmen as **Academic Mentor** of **BSW** for the course **Introduction to Computer Science**.
- Taught elementary Mathematics, Science, and English to 30+ children in a slum under **National Service Scheme**.