KALYANI GOVERNMENT ENGINEERING COLLEGE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Presents

prisma

IDEAS
CODE
CREATIVITY
IMPACT

AI
DATA
ALGORITHMS
INNOVATION
TOMORROW



Same Humanity. New Possibilities.

EDITION 3



President

Dr.Sourabh Kumar Das
Principal,

Kalyani Government Engineering College

Secretary

Dr. Kousik Dasgupta
Head of Department,

Computer Science & Engineering



Faculty Memebers

Dr. Kousik Dasgupta

Prof. Malay Kumar Pakhira

Prof. Swapan Kumar Mondal

Dr. Sudip Kumar Adhikary

Dr. Anup Kumar Biswas

Mr. Pramit Brata Chandra

Mrs. Manju Biswas

Dr. Arindrajit Pal

Dr. Subbhankar Ghosh

Dr. Arun Kumar Sadhu

Dr. Tanmoy Chakraborty

Mr. Deepak Kumar Jha

Mr. Kuntal Bhowmick

Mr. Pranab Ranjan Das

From the Desk of President

Dear CSE Department,

It is with great pleasure that I extend my heartfelt greetings to the talented members of our Computer Science and Engineering family. As we embark on another academic journey, I am filled with pride and enthusiasm for the remarkable achievements and unwavering dedication that define our CSE department.

In the ever-evolving landscape of technology, our CSE department continues to be a beacon of excellence. Our students, faculty, and staff contribute tirelessly to the dynamic field of computer science, pushing boundaries and shaping the future.

The collective brilliance within these walls is not only inspiring but also a testament to our commitment to fostering innovation, collaboration, and a relentless pursuit of knowledge.

I commend the initiative to showcase the department's achievements through this magazine, "Prisma 3.0". Let the pages within be a reflection of our shared journey, celebrating successes, acknowledging challenges, and highlighting the collective spirit that defines us. I encourage everyone to actively participate and contribute to making this edition a true representation of the outstanding work emanating from our CSE family.

Wishing you all a year filled with intellectual growth, innovative endeavors, and collaborative triumphs.

Best Regards,
Dr. Sourabh Kumar Das
President,
Prisma
Kalyani Government Engineering College

From the Desk of Secretary

It gives me immense pleasure to welcome you to the latest edition of our departmental magazine “Prisma 3.0”—a vibrant celebrate of creativity, thought, and artistic expression beyond the realm of code.

Technology is a powerful tool for transformation, but it is human curiosity, empathy, and creativity that give those tools true purpose. As future computer scientists, your logical rigor builds the foundation of progress, but your character and values define the ultimate impact of your innovations. True engineering excellence goes beyond writing flawless algorithms; it lies in solving real human problems with integrity and heart.

This magazine serves as a canvas to showcase your unique perspectives, literary talents, and passions outside the classroom and laboratory. I encourage each of you to continuously step out of your comfort zones, embrace failure as a master teacher, and develop the adaptability and ethical leadership needed to navigate an ever-evolving world.

May your vision remain remarkably bold, your passion for learning stay burning bright, and your creative endeavors continue to inspire those around you. Keep building, keep exploring, and keep shaping a future that reflects the finest of human potential.

I congratulate the editorial team and all contributors for bringing this wonderful publication to life. Wishing you all an inspiring read!

Warm Regards,
Dr. Kousik Dasgupta,
Secretary,
Prisma
Kalyani Government Engineering College



Editor-in-chief

Debayan Bhattacharya
4th year CSE



Content Lead

Maryada Tudu
4th year CSE



Technical Lead

Sayan Chatterjee
4th year CSE

STUDENTS' MAGAZINE COMMITTEE MEMBERS

4th YEAR

Ayush Mandal
Ishita Mukherjee
Deepjyoti Barua
Anupam Mandal

3rd YEAR

Chayan Samanta
Syed Mosharaf Hossain
Dipanwita Mandal
Sunita Singha Mahapatra
Samriddha Das
Supriti Barua
Ritisha seal

2nd YEAR

Rupam Sarkar
Sunetra Bar
Anubhab Mondal
Garbita Chowdhury
Archisman Halder
Mainak Das
Sk Asif Hussain

INDEX



Technical Articles 8-25



Stories 26-35



Technical Essays 36-43



Poems 44-50



Photography 51-56



Drawing 57-61



TECHNICAL ARTICLES

The War on Hallucinations

Beyond Stochastic Parrots: Hierarchical Verification and Self-Correction in LLMs

Introduction:The Mathematical Inevitability of Hallucinations

Large Language Models (LLMs) have irrevocably transitioned from experimental prototypes into mission-critical infrastructure. As these intelligent systems are deployed in high-stakes environments—ranging from automated software engineering to autonomous financial execution—the engineering mandate has shifted from scaling parameter counts to enforcing probabilistic reliability.

What Is an AI Hallucination?

In Plain English: An AI hallucination happens when an AI model confidently makes something up. It doesn't say "I don't know"—it presents fabricated facts, invented statistics, fake legal cases, or non-existent medical studies as if they were real. The response sounds authoritative and reads perfectly. That is exactly what makes it so dangerous.

Technical Definition: Hallucination refers to generated output that is not grounded in the provided input data or factual reality. There are two primary types:

Intrinsic hallucination (faithfulness hallucination): The model contradicts information explicitly provided in its source material.

For example, during summarisation, it adds facts not present in the original document.

Extrinsic hallucination (factuality hallucination):

The model generates information that cannot be verified against any known source—it invents facts, citations, statistics, or events from scratch.

Why It Happens?

Large Language Models (LLMs) are fundamentally prediction engines, not knowledge bases. They generate text by predicting the most statistically likely next word based on patterns learned from training data. They do not understand truth, they predict plausibility. When the model encounters a gap in its training data or faces an ambiguous query, the mathematics of its softmax function force it to guess the most plausible-sounding continuation, bridging gaps in logic with seamless fiction.For computer science engineers, the mandate is clear: we can no longer treat these hallucinations as traditional software "bugs" that can be patched. Because this stochastic behaviour is mathematically inevitable, absolute reliability must be engineered through external architectural oversight.

Engineering Reliability:

Reflection and Hierarchical Verification

Contemporary research has moved decisively away from single-pass inference towards multi-layered verification architectures. Instead of relying on a single, flawed generative pass, we deploy both internal and external auditing mechanisms to force the model to reason accurately.

Intrinsic Reflection (Self-Correction)

The first line of defence is intrinsic reflection. Instead of outputting its initial statistical prediction, the model is forced into an iterative self-critique loop. It drafts a response in a hidden scratchpad, and a secondary system prompt instructs the model to critique its own logic checking for contradictions and unsupported claims before delivering the final output.

Extrinsic Verification (Multi-Model Orchestration) :

However, a model cannot always detect its own blind spots. When intrinsic reflection fails, the state-of-the-art relies on extrinsic hierarchical verification, known as the Proposer-Verifier-Synthesiser pattern. This mirrors dual-process theory in cognitive psychology by dividing tasks into intuitive generation and analytical reasoning.

Proposer : Generates a high-entropy draft, exploring the latent space with deliberate creative latitude. It prioritises recall and possibility exploration over strict factual adherence.

Verifier : A deterministic, specialised auditing layer. Frameworks like the Verifiable Iterative Refinement Framework (VIRF) utilise formal logic (such as Web Ontology Language or OWL rules) to parse and evaluate proposed actions.

Synthesiser: Integrates only the verified claims into a final output, ensuring the response is logically coherent and explicitly auditable.

The Power of Pedagogical Feedback : Crucially, these systems work together dynamically. Modern Verifiers don't merely reject incorrect outputs; they generate pedagogical feedback. For example, if the Proposer suggests using an $O(n^2)$ sorting algorithm for a massive dataset, the Verifier doesn't just return "Error." It outputs a logical proof: "Constraint violation: $O(n^2)$ complexity on large data causes timeout. Memory limit exceeded. Suggestion: Implement external merge sort." This feedback is fed directly back into the Proposer's intrinsic reflection loop, allowing the system to self-correct and generate a mathematically sound solution.

Empirical benchmarks highlight the success of this approach. In tests mirroring Anthropic's Constitutional AI, injecting a Verifier reduced hallucinated citations by over 50% while maintaining high recall, proving that separating generation from validation is fundamentally superior to single-pass prompting.

RAG 2.0 : Graph-Based Contextual Anchoring

While reflection fixes logical errors, factual retrieval requires structural anchoring. Traditional Retrieval-Augmented Generation (RAG) struggles with "semantic drift" across complex queries.

GraphRAG addresses this by representing knowledge as structured Knowledge Graphs rather than disconnected vector embeddings. Using advanced clustering algorithms, GraphRAG allows models to traverse explicit relational edges, creating an auditable reasoning path. While GraphRAG yields a massive 50–70% improvement in global summarisation and excels at multi-hop reasoning, it introduces significant trade-offs compared to traditional vector RAG: high index construction costs, a 2.3× increase in query latency, and up to a 13.4% accuracy degradation on highly specific factual queries.Furthermore, GraphRAG is vulnerable to poor metadata governance; noisy source data creates convoluted graph topologies. To fix this, engineers are adopting programmatic optimisation frameworks like DSPy (Declarative Self-Improving Python). Instead of relying on brittle manual prompts, DSPy algorithmically tunes smaller models to perform strictly typed, schema-compliant entity extraction. This ensures only semantically valid relationships enter the knowledge graph, which drastically reduce hallucination rates.The Future: Reason-to-Code Architectures.

To achieve zero-defect reliability in algorithmic domains, systems engineers have formalised the Reason-to-Code loop. Acknowledging that neural networks are fundamentally unsuited for precise arithmetic due to floating-point attention mechanism

the model is forced to output formal code rather than a direct numerical answer.

This code executes in a secure, sandboxed environment. The interpreter acts as a deterministic oracle, feeding the verified computational result back to the model for final natural language synthesis. By bypassing probabilistic attention heads in favour of deterministic CPU execution, we neutralise the mathematical inevitability of hallucination.

Conclusion

Recent theoretical work reframes the "stochastic parrot" critique by defining semantic grounding not as an internal hidden state, but as a verifiable predictive relationship between symbols and observable phenomena.

This perspective aligns perfectly with modern architecture. We are no longer treating hallucination mitigation as a superficial patch. Through intrinsic reflection, hierarchical verification, DSPy-optimised GraphRAG, and Reason-to-Code loops, we are fundamentally evolving LLMs from probabilistic text generators into verifiable cognitive engines. For the systems architects of tomorrow, this represents the true frontier.

~Manab Biswas, 4th year, CSE

Your OS is Holding You Back

It's Time to Switch to Linux

There is an ongoing revolution in the computing space, and most individuals are completely unaware of it. Individuals are still routing updates on Windows and utilizing Apple's excessively secure walled garden, but millions of end users, including students, artists and developers, as well as large corporations, are making the migration to Linux. In 2026, the justification for switching has never been stronger. Whether you are a user that only uses the internet to browse or watch YouTube videos or you are a developer that stays up all night compiling the kernel, Linux has an incredible amount to offer you. Here are some things that make now the right time to switch.

Freedom From the Subscription Trap

TMicrosoft has transitioned to a fee-based subscription system for Windows 11, while macOS is effectively locked into very expensive Apple equipment. Linux is a free operating system; this means that it is completely free of monetary cost and it is also a philosophy. Both the operating system itself (along with thousands of applications) and an extensive tool set for building applications can be downloaded and installed without any monetary cost.

Moving to Linux is life-changing for everyone including students and small business owners.You can install (using a DVD, USB drive or hard drive connection) full-featured Linux distribution on any computer and it will be a functioning and complete, professional desktop environment at no cost.

Linux Breathes Life Into Old Hardware

Windows 11 will not work on millions of perfectly good computers. Instead of telling you that you must upgrade your machine AND purchase new hardware, you can simply install a lightweight version of Linux (like Linux Mint or Xubuntu) on your current computer and get a fast, modern computing experience. You can take a computer that is 10 years old and would run slow with Windows and have it now run fast and smooth with quick boot times and no extraneous applications in the background. Not only does this save the environment, it also saves money.

Privacy Is Built In, Not Bolted On

By default, Microsoft collects usage data, including telemetry (keystrokes), usage patterns, browser habits, application data, etc. Even with privacy settings configured to collect the least amount of data, Windows will still collect some amount of data from you. The Linux operating system does not collect any information.Linux is open source, meaning that every line of code can be reviewed by anyone. When data is becoming more and more like oil, it is important to have an operating system that gives you privacy, and not paranoia, when it comes to protecting your data. When it comes to operating systems, Linux is the only major operating system that you can be absolutely sure that your data will be kept safe.

The Developer's Natural Habitat

For all those working with technology, particularly in software development or related fields (such as Computer Science), using Linux is virtually unavoidable. Around 70% of the earth's web servers run on Linux, as do all forms of cloud computing, supercomputers, etc. You will already be ahead when you apply for industry positions by knowing how to use Linux.

Linux has:

- Bash/Zsh/Fish Terminal (very feature rich)
- Out of the box - supports all programming languages natively - Python, Node.js, C++, C#, Rust, Go, and nearly all others.
- Docker, Kubernetes, and all other tools for containerization with no workarounds
- SSH, cron jobs, and system scripting are installed and preconfigured as part of your Linux install
- All official Linux package managers install software within seconds Microsoft has acknowledged that all developers need to run Linux by offering Windows Subsystem for Linux (WSL).

If you are running a virtual instance of Linux from within a Windows system, why wouldn't you just be using Linux?

Casual Users: Linux Is Easier Than You Think

Historically, it was widely thought (based upon truths of the past) that Linux would never become a product comfortable for those who have no computer-related background.Thankfully this is no longer true. There are many great modern versions of Linux that have been built to provide point-and-click usage without almost any technical effort from you.

IThe major official variations are considered to be Ubuntu, Pop!_OS, and Linux Mint. While they only require 20 minutes of my time to install, I find that the vast majority of hardware (WIFI, Bluetooth, Web Cameras) will be automatically working upon first boot. When it comes to doing everyday activities on the Internet (Web Browsing- Firefox, Google Chrome, Watching Television Over The Internet, Word Processing- Document Accessibility Via LibreOffice, Photographs/Photo Management, etc.). You can do almost everything via Linux without any experience or problems. In contrast to Microsoft Windows, you will never have to experience mandatory Updates that may require 30 minutes of your time or having your computer crash to restart after an update. Additionally, you will not have any advertising or solicitations within the Start Menu.

Security That Does Not Require an Antivirus

Linux has many layers of security built into its foundation as compared to other operating systems. The file permission structure means that if a piece of malware infects your PC, it cannot alter any system-level files because of not having 'root' access granted to that piece of malware. In terms of viruses, it is virtually impossible to find one that is only targeting desktop/Laptops using Linux. This means you will never need to purchase, configure, or maintain antivirus software while utilizing Linux. Security patches for Linux OS are released on the same day they are discovered, eliminating the waiting period associated with Microsoft's 'Tuesday Patch Cycle'. Therefore, you do not have to wait for Microsoft to fix a zero-day vulnerability on your system.

A Community That Actually Helps

The Linux community is one of the "possibly" (discussed above) most active and helpful communities when it comes to technology. There are numerous forums, such as Ask Ubuntu and the Arch Wiki, where many of the problems that you will run into will have been solved and documented, so you will not be alone. The Linux community is so active that instead of relying on a support team of a corporation and having to wait for a ticket to be answered, you will usually get a response from the community within a matter of minutes. The community also drives the evolution of Linux based on what the users of Linux really want.

The Switch Is Easier Than the Fear

The biggest barrier to switching to Linux is not technical — it is psychological. The fear of the unfamiliar keeps people on operating systems that surveil them, slow them down, and extract money from them. But the learning curve is far gentler than it was even five years ago.
Start with Linux Mint or Ubuntu in a virtual machine. Try it alongside your current OS for a week. You will likely find that you do not miss Windows — you will wonder why you waited.

The operating system you choose is not just a tool — it is a statement about what you value: freedom over restriction, transparency over surveillance, and community over corporate control. Linux offers all three. The question is not whether you should switch. The question is why you haven't already.

~Subhraditya Ghosh, 2nd Year, CSE

Hashing: Turning Big Things into Small Codes

How a tiny code quietly protects your passwords, checks your downloads and finds yourdata in a flash

What is hashing?

Think about the cloakroom at a function, or the shoe stand outside a temple. You hand over your bag, they give you a small token, and later you just show the token to get your bag back. Hashing in computers follows a similar idea. Something big or complicated goes in, and a short code comes out. That code is called a hash.

The tool that makes the code is called a hash function. Give it a single word, a photo or a full movie, and the code it returns will always be the same length. Give it the same input again tomorrow and you get exactly the same code.

There is one more thing to know. You cannot take a hash and turn it back into the original data. It only works in one direction, which is why hashing is used to check things and not to hide them.

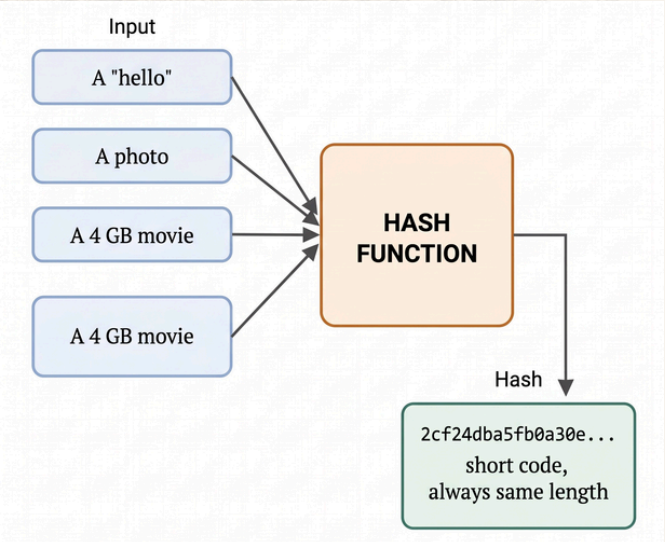


Figure 1: Different inputs go into the hash function and a fixed-size code comes out.

How a hash function behaves

The same input always gives the same hash.

Any size of input gives the same size of output. SHA-256, a popular hash function, always gives 64 characters.

Change even one letter of the input and the whole hash looks different.

You cannot go backwards from the hash to the original data.

It is quick to calculate, even for big files.

The table shows real SHA-256 hashes of four small inputs. Only the first 24 of the 64 characters are printed. Look at how little 'hello' and 'Hello' have in common, even though only
one letter is different.

Hash tables and collisions

Programs store things like contacts and usernames in a structure called a hash table. Picture a cupboard with numbered drawers. The hash function looks at a name and tells the computer which drawer to open, so it goes straight there instead of checking every drawer one by one. Sometimes two names get the same drawer number. They should not be used where security matters. SHA-256 is still considered safe.

This is called a collision, and it is bound to happen now and then because there are far more possible names than drawers. A common fix is chaining, where a drawer keeps a small list of everything that landed in it, like Kabir behind Arjun above. Another fix is open addressing, where the program simply moves on to the next empty drawer. Collisions also matter for security. Older functions such as MD5 and SHA-1 have been broken, because researchers found ways to create two different inputs with the same hash on purpose.

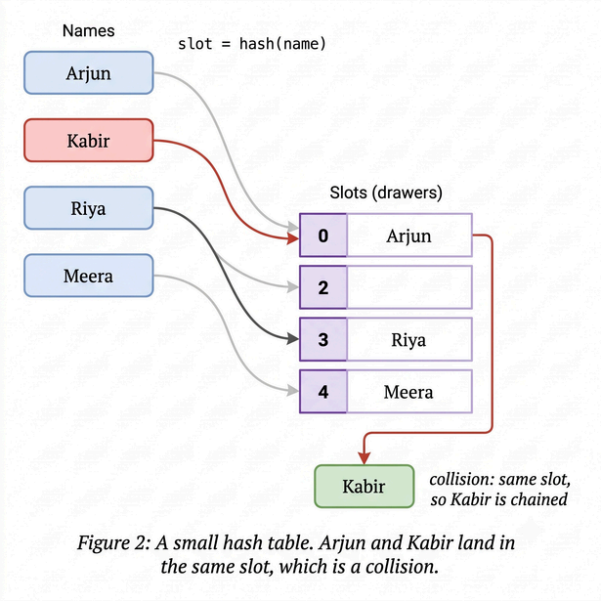


Figure 2: A small hash table. Arjun and Kabir land in the same slot, which is a collision.

Where hashing is used

• **Passwords:** A well-built website stores the hash of your password, not the password. When you log in, it hashes what you typed and compares the two. If someone steals the database, they get hashes and not the real passwords.

• **Downloads:** Some websites publish the hash of a file. You hash your downloaded copy, and if both match, the file was not changed on the way.

• **Cloud storage:** Two identical files have the same hash, so a service can notice that and keep only one copy.

• **Git:** Every commit gets an ID that is really a hash, which is why commits look like long strings of letters and numbers.

• **Blockchain:** Each block stores the hash of the block before it. Changing an old record would break every block after it.

Things to keep in mind

Never store passwords as plain text. Use a purpose-built password hash such as bcrypt, scrypt or Argon2, which are deliberately slow so guessing is expensive for an attacker. Also add a salt, which is a random piece of text mixed into the password before hashing. With a salt, two people who pick the same password still end up with different hashes. For checking files, SHA-256 or better is a good choice.

Wrapping up

Hashing sits quietly behind a lot of what we do online: logging in, downloading a file, saving a contact. The idea underneath is simple. Shrink anything into a short fixed-size fingerprint, then use that fingerprint to find, compare and verify. Once you know what to look for, you start spotting it everywhere, from a cloakroom token to the password box on your screen.

~ Mainak Das, 2nd Yr, CSE

Build, Break, Learn

- The Joy of the Home Lab

A home lab is more than a collection of devices and wires. It is a personal space for experimentation, discovery, and hands-on technical learning. For beginners, it offers something that books and lectures often cannot: the chance to build real systems, make real mistakes, and understand how technology behaves in practice.

At its core, a home lab is a small environment created for testing, learning, and exploration. It can begin with simple equipment and grow over time as curiosity grows. What makes it powerful is not its size, but its purpose. It gives learners a controlled space where they can install systems, connect services, observe behavior, and improve what they build.

The journey usually starts with a question. How do systems communicate? How are applications deployed? What happens when a service fails? How is a network organized? A home lab turns these questions into experiments. Instead of stopping at theory, the learner begins to test ideas directly. One setup leads to another, and soon the lab becomes a living workshop of technical thought.
This is where much of the excitement comes from. A home lab makes learning active. It invites trial, error, adjustment, and success. Something may fail, but that failure becomes part of the lesson. A misconfigured service teaches troubleshooting. A slow system teaches resource management. A broken connection teaches the logic of networking. Each problem solved adds not only knowledge, but confidence.

The learning is wide and practical. A person working in a home lab may begin to understand system setup, storage planning, network behavior, access control automation, recovery, and performance monitoring. More importantly, the learner develops habits that matter deeply in engineering: patience, observation, structured thinking, and the ability to improve a design step by step.

Yet the real charm of a home lab lies in its sense of ownership. There is genuine satisfaction in building something small and watching it work. There is fun in testing ideas freely, in changing configurations, in making the setup better than it was the day before. The lab becomes a technical playground, but also a creative one. It rewards curiosity and turns learning into a personal adventure.

For an engineer, this kind of environment feels natural. Engineering is rarely just about knowing facts; it is about understanding systems through design, testing, and refinement. A home lab supports exactly that mindset. It helps transform abstract concepts into visible results and gives learners a place to practice the deeper craft of engineering. In the end, a home lab is not simply a setup for machines. It is a workshop for building skill, confidence, and technical imagination.

~ Dwaipayan Datta, 3rd Yr, CSE

The Frontiers of Artificial Intelligence & Data Science

Artificial Intelligence (AI) and Data Science have transitioned from niche academic disciplines into foundational pillars of global technological infrastructure. In 2026, these fields stand at an unprecedented convergence point: advances in large language models, autonomous agents, edge computing, and multimodal learning are reshaping industries from healthcare to finance, manufacturing to education. This report surveys the current frontiers of AI and Data Science, examines key breakthroughs, outlines persistent challenges, and forecasts the trajectory of these disciplines over the next decade.

Introduction

The evolution of AI has been marked by cyclical periods of optimism and stagnation, commonly referred to as "AI winters." However, the advent of transformer-based architectures in 2017, combined with exponentially growing computational resources and vast data availability, has ushered in a sustained era of breakthroughs. Data Science, as a complementary discipline, provides the methodological toolkit to extract actionable insights from structured and unstructured data at scale.

Together, AI and Data Science are no longer aspirational technologies. They are operational realities embedded in search engines, medical diagnostics, autonomous vehicles, climate modelling, and financial systems. Understanding their frontiers is essential for researchers, policymakers, engineers, and business leaders alike.

1. Key Frontiers in Artificial Intelligence

1.1. Large Language Models and Foundation Models

Large Language Models (LLMs) such as GPT-4, Claude, Gemini, and LLaMA have demonstrated emergent capabilities that extend well beyond text generation — including logical reasoning, code synthesis, mathematical problem-solving, and instruction-following across diverse domains. The paradigm of "foundation models" — large pre-trained models fine-tuned for specific downstream tasks — has become the dominant approach in NLP, vision, and multimodal systems. Key research frontiers include improving reasoning reliability, reducing hallucination, enabling long-context comprehension, and scaling to trillion-parameter architectures efficiently, while simultaneously developing highly optimized Small Language Models (SLMs) for localized, on-device inference.

1.2. Autonomous AI Agents

Agentic AI systems — capable of planning, tool use, and executing multi-step tasks with minimal human oversight — represent one of the most consequential frontiers in applied AI. Frameworks such as ReAct, AutoGPT, and Anthropic's Computer Use have demonstrated that LLMs can orchestrate complex workflows involving web browsing, code execution, API calls, and file management. The challenge lies in ensuring robustness, safety, and alignment in open-ended environments where errors can cascade unpredictably.

1.3. Multimodal and Embodied AI

Modern AI systems increasingly integrate text, images, audio, video, and sensor data into unified architectures. Multimodal models such as GPT-4V,

Gemini Ultra, and Florence-2 can reason across modalities simultaneously. Embodied AI — systems that perceive and act within physical or simulated environments — is advancing rapidly through robotic manipulation, autonomous navigation, and simulation-to-real transfer techniques. This frontier is critical for robotics, augmented reality, and industrial automation.

1.4. AI Safety and Alignment

As AI systems grow more capable, the alignment problem — ensuring that AI behaviour reliably reflects human values and intentions — has moved to the forefront of both academic and institutional concern. Techniques such as Reinforcement Learning from Human Feedback (RLHF), Constitutional AI, interpretability research, and formal verification are being actively developed. Organizations including Anthropic, DeepMind, and OpenAI have dedicated safety research teams, and regulatory frameworks in the EU, US, and UK are beginning to codify safety standards for high-capability AI systems.

2. Key Frontiers in Data Science

2.1. Real-Time and Streaming Analytics

The proliferation of IoT devices, financial tick data, and social media streams has necessitated data pipelines capable of processing and analysing data at millisecond latency. Technologies such as Apache Kafka, Apache Flink, and cloud-native stream processing services have matured considerably. The frontier now involves integrating ML inference directly into streaming pipelines, enabling real-time fraud detection, predictive maintenance, and dynamic content personalization.

2.2. AutoML and Democratization of Data Science

Automated Machine Learning (AutoML) platforms are reducing the barrier to model development for non-experts, while allowing seasoned data scientists to rapidly establish baseline models and accelerate the development lifecycle. Coupled with AI-assisted data wrangling and feature engineering, these tools are democratizing access to predictive modelling. The frontier involves not only automating model selection and hyperparameter tuning, but also automating the full data science lifecycle including hypothesis generation, experiment design, and insight narration.

2.3. Federated Learning and Privacy-Preserving AI

Data privacy regulations such as GDPR and HIPAA constrain the centralization of sensitive datasets. Federated Learning addresses this by training models locally on distributed devices and aggregating only model updates — not raw data — on a central server. Differential privacy techniques further ensure that individual data points cannot be reverse-engineered from model outputs. These approaches are critical in healthcare, finance, and government applications where data sovereignty is paramount.

3. Persistent Challenges

Despite rapid progress, several fundamental challenges remain unresolved:

- Interpretability: Despite emerging breakthroughs in mechanistic interpretability, deep learning models remain largely opaque, limiting trust and adoption in strictly regulated domains.
- Data Quality and Bias: AI systems trained on biased or incomplete data perpetuate and amplify societal inequalities.
- Computational Costs: Training frontier models consumes enormous energy and resources, raising sustainability concerns.
- Robustness and Generalization: Models often fail catastrophically on out-of-distribution inputs or adversarial perturbations.
- Governance: The rapid pace of AI deployment outstrips the capacity of regulatory and legal frameworks to keep pace.

4. Outlook and Future Directions

The next decade of AI and Data Science will likely be defined by the convergence of powerful generalist AI systems with domain-specific expertise, embedded into physical and digital infrastructure at scale. Key developments to anticipate include reasoning-capable AI agents deployed in scientific discovery and drug development, neuromorphic hardware enabling energy-efficient on-device AI, and the emergence of human-AI collaborative workflows as the dominant paradigm in knowledge work.

Governance frameworks, safety research, and interdisciplinary collaboration will be as important as technical progress in shaping whether these technologies deliver broadly beneficial outcomes.

5. Conclusion

AI and Data Science are advancing at a pace that presents both extraordinary opportunity and substantial risk. The frontiers discussed in this report — from foundation models and autonomous agents to federated learning and real-time analytics — represent areas where continued investment and rigorous research are essential.

Stakeholders across academia, industry, and government must work in concert to ensure that the development of these technologies is guided by principles of safety, equity, and long-term societal benefit.

~Sumit Sinha, 2nd Year, CSE

Impacts Of Artificial Intelligence(AI) In Indian Job Markets

~Pramit Brata Chanda, Faculty, CSE Department

Today in global economy Artificial Intelligence(AI) makes a crucial impact throughout the world in various domains. So various service sectors like IT, Manufacturing, Operations, Finance, Business Analytics are already very much affected with AI and their related sub-domain. So, After 2022, rapid uses of one technology(AI Tools) whatever we use in research area open throughout the world as AI applies in every particular sectors. Whatever efficiency we achieve after study in our school, college life or working life, using various experience and knowledge, AI based tools can perform all of those job within 5-10 seconds. Also, AI tutors can explain concepts, answer questions, provide practice exercises, and give instant feedback. AI can create lesson plans, quizzes, worksheets, presentations and study materials.AI can automate attendance records, scheduling, student communications, report generation. This could reduce demand for some basic tutoring and coaching roles, particularly for standardized subjects. So, the effects of this globalization of AI can make impacts towards IT sector also.

In the period of 2012-13 to 2022-23 the Indian economy increases growth more than 3 to 4 times on basis of Computer And Information Services after Trading and Transport Services. After that period, AI gives higher impact in overall our country's job market specially on Entry Level jobs all over India. Entry level jobs depends on new functionality, description of inputs, outputs, Building Algorithm, Code access rights. Also uses of Agentic AI can affects job markets more than 50% overall in India within 6-8 years. Also, previously our concepts was learning HTML, Java, Web Technology, DataBase of catching a healthy jobs in IT Sector. But now Large Language Model(LLM), Generative AI Based Framework change the overall IT Industry Job markets quickly and also become the hard reality of Todays IT Sector of India.HTML, Java, Web Technology, DataBase of catching a healthy jobs in IT Sector. But now Large Language Model(LLM), Generative AI Based Framework change the overall IT Industry Job markets quickly and also become the hard reality of Todays IT Sector of India.

AI can reduce demand for some jobs, but it does not necessarily mean that AI will reduce the total number of jobs overall as given below:-

Job Displacement: AI can automate repetitive or predictable tasks, such as data entry, basic customer support, document processing, and some routine coding.

Job Transformation: Many jobs may not disappear, but workers may use AI to perform parts of their work faster. For example, accountants, designers, programmers, teachers, and analysts may increasingly work alongside AI.

New Jobs: AI also creates demand for new roles, including AI engineers, data specialists, AI safety professionals, automation specialists, and people who manage AI systems.

Productivity Effect: If AI allows companies to produce more with fewer workers, employment can fall in some occupations. But lower costs and new products can also create additional economic activity and jobs.

Skills Matter: Workers whose jobs contain highly automatable tasks may face more disruption, while people who combine AI skills with communication, creativity, management, or specialized knowledge may have new opportunities.

Reduction In Entry-level Opportunities: This is particularly important for young workers. Jobs such as junior analysts, customer-service representatives and basic content creators can provide experience that leads to higher-level positions. If AI performs these tasks, fewer entry-level positions may be available.

Skill Gap and Inequality: Workers who know how to use AI may become more productive, while workers without digital/AI skills may struggle. This can increase the gap between highly skilled and less-skilled workers.

Workplace Surveillance: AI can monitor employee productivity, calls, keystrokes, customer interactions and working patterns. Excessive monitoring can create stress and reduce employees' sense of autonomy.

Loss of Human Interaction: Customer service increasingly handled by chatbots or voice AI can make interactions less personal. Complex or emotionally sensitive problems may be difficult for automated systems to handle appropriately.

There are already documented examples where AI or AI-powered automation has reduced the need for certain roles or led companies to cut jobs. A few examples shown below:-

Customer Service

AI chatbots can handle common questions, refunds, appointment scheduling, and basic troubleshooting. Companies can therefore need fewer human agents for routine queries.

Call Centers

AI voice systems can conduct simple conversations, identify customer needs, and route calls. This can reduce demand for workers handling highly repetitive calls.

Content Writing

Generative AI can produce product descriptions, simple news summaries, marketing copy, and other routine text. Some companies have reduced freelance or entry-level writing work as a result.

Translation

Machine translation can handle large volumes of straightforward translation. Human translators are still important for specialized, legal, literary, or highly nuanced work, but routine translation tasks can require fewer people.

Data Entry and Administrative Work

AI-powered document processing can extract information from invoices, forms, receipts, and emails. This reduces the amount of manual data-entry work required.

Software Development

AI coding tools can generate code, tests, documentation, and debugging suggestions. This doesn't mean programmers disappear, but a smaller team may be able to accomplish some tasks that previously required more junior developers.

Graphic Design

Generative AI can create basic advertisements, social-media images, logos, and illustrations. This can reduce some demand for simple, repetitive design assignments.

402: Payment Required — How AI Agents Learned to Pay Their Own Way

Dr. Kousik Dasgupta, HOD, CSE Department

For thirty years, the online economy has run on one quiet assumption: the customer is a person. A person browses the website. A person types in a card number or scans a QR code. A person clicks "Buy".

AI agents are starting to break that assumption. Cloudflare recently reported that automated systems now generate around 57.5% of HTTP requests for web content. AI tools are also growing up. They are no longer just chatbots that answer questions; they are becoming agents that draw up plans, act on them, and check whether things worked out. Yet the way we pay for things online is still built entirely around humans. Figure 1 shows how differently the two payment journeys play out.

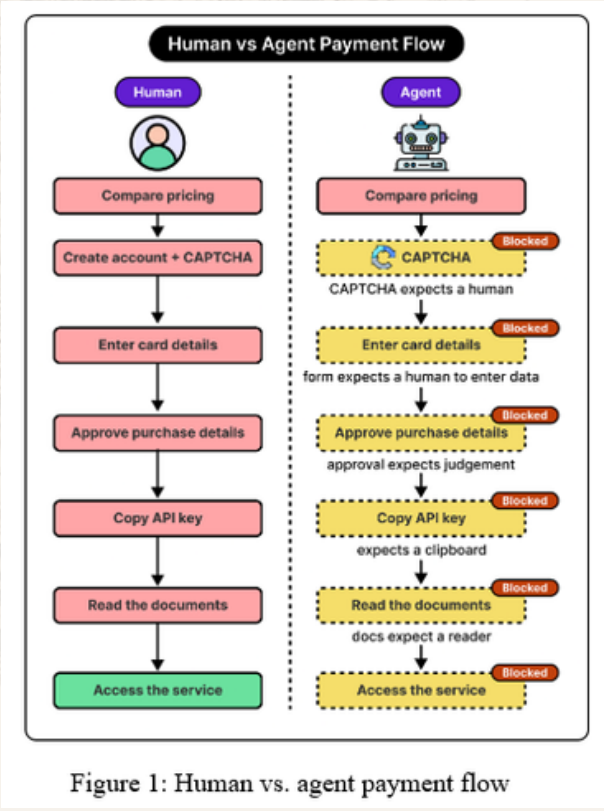


Figure 1: Human vs. agent payment flow

If ChatGPT is going to be your customer, the internet needs a new kind of protocol. That is where the Machine Payments Protocol (MPP) [1] comes in. MPP was co-authored by Stripe and Tempo and launched on 18 March 2026. Its core specification has been published as the Payment HTTP Authentication Scheme and submitted to the IETF standards track, the same body that looks after HTTP itself.

Before any money changes hands, the two sides have to agree on a few things: what the service costs, what counts as payment, and what proof of payment should look like.

People settle all of this at a glance, usually without even noticing. MPP spells out each of these steps and gives each one a name:

- The Challenge is what the server asks for (the payment) when someone requests a service.
- The Credential is what the client sends back as proof that it has paid.
- The Receipt is what the server hands back along with the service.
- These three objects [2] sit at the heart of the protocol, alongside three payment intents: charge, session, and subscription. Figure 2 shows how they fit together.

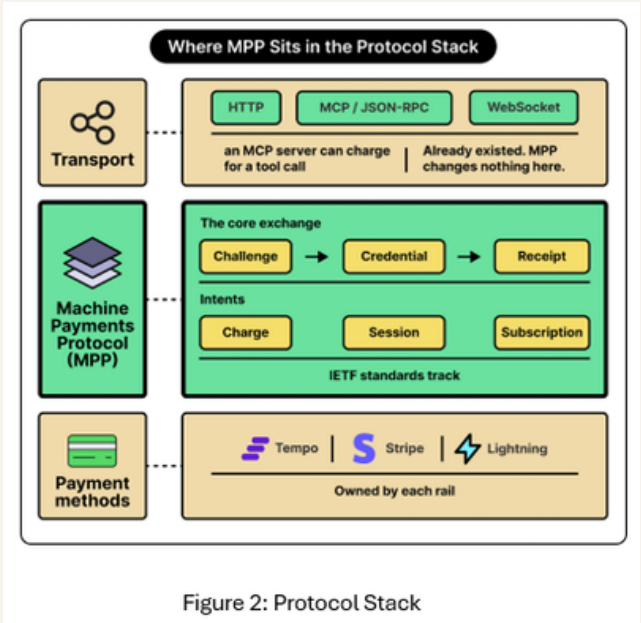


Figure 2: Protocol Stack

Here's the catch. Every website presents its payment details a little differently. For people, that hardly matters, because we're good at working things out. Software isn't. It needs the terms in the same place and in the same format every time so it can read them reliably. So MPP puts them in HTTP headers. The server states its terms in a WWW-Authenticate: Payment header, the client answers with proof in an Authorization: Payment header, and the server confirms with a Payment-Receipt header. The Challenge itself carries an ID, the amount, the currency, who gets paid, the payment method, and how long the offer stays valid. The ambition behind all this is simple: MPP wants to become the common language of payments on the internet. Any service that speaks MPP can sell to any agent, and any agent that speaks MPP can buy from any service.

How Does MPP Work? Reviving HTTP 402

MPP brings back HTTP 402 "Payment Required" [3-5], a status code that has sat mostly unused since the early days of the web. When an agent calls your API or service, you reply with a 402 and the payment details. The agent pays, sends the request again, and gets what it asked for along with a receipt. Figure 3 shows the full flow, step by step:

1.The agent asks for access to a service or an API.

2.The server names its price. It replies with 402 Payment Required and puts its terms in a WWW-Authenticate: Payment header. That header carries the Challenge: a unique ID, the payment method, the intent, and an expiry, plus an encoded payment request with the amount, the currency, and the recipient's address.

3.The agent reads the terms. There's no sign-up, no form to fill in, and no API key to buy. The agent simply checks the amount, the recipient, the currency, and the validity window in the signed payload.

4.The agent authorizes the payment, but only within limits set in advance. Crucially, those limits live in the signing key, not in the agent's reasoning. It is a delegated key with a spending cap per period, an expiry date, a list of approved recipients, and a defined scope. There is usually one key per deployment, and each can be revoked on its own, so a runaway agent can't spend beyond its limit.

5.The agent proves it has paid. It sends the same request again, this time with an Authorization: Payment header containing a Credential. The Credential holds the Challenge ID and a payment payload, which could be a signed transaction, a paid invoice, or a card token. Because credentials can spend real money, they are treated as bearer instruments: servers and intermediaries never log them or echo them back in error messages.

6.The server delivers. It checks the proof against the payment rail, returns what was requested (search results, in the example in Figure 3), and attaches a Payment-Receipt header confirming that the payment has settled.

Adoption:

MPP has been proposed to the IETF, and more than 100 services signed on at launch, including Anthropic, OpenAI, Shopify, Alchemy, and Dune Analytics.

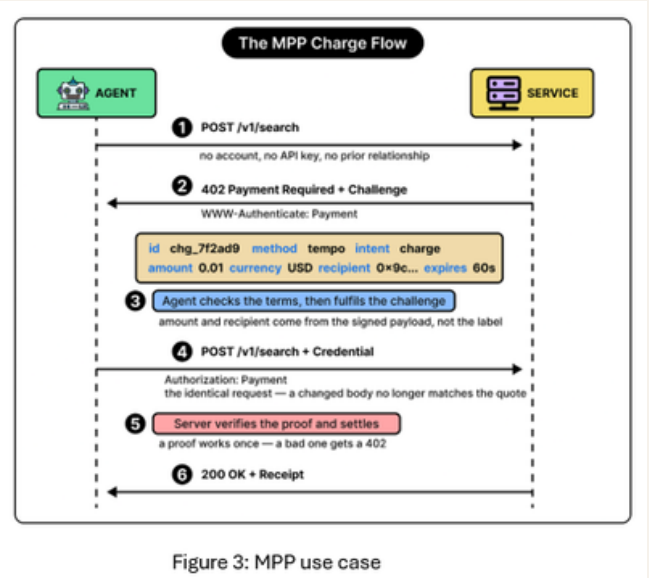


Figure 3: MPP use case

For Stripe merchants, agent payments show up in the dashboard like any other transaction and settle into the same balance on the usual payout schedule.

Your Next Customer Might Be an Agent:

MPP has been live since March 2026, and the volume is still modest. But the numbers aren't the real story. What matters is that a new kind of customer has just walked through the door, and it doesn't need a checkout page.

References:

[1] J. Weinstein and S. Kaliski, Introducing the Machine Payments Protocol, https://stripe.com/blog/machine-payments-protocol (accessed Oct. 4, 2026).
[2] "MPP," MPP | Stripe Documentation, https://docs.stripe.com/payments/machine/mpp?utm_source=substack&utm_medium=email (accessed Oct. 4, 2026).
[3] "HTTP 402," MPP - Machine Payments Protocol, https://substack.com/redirect/7af61674-173a-40fb-9ded-bf2d9dff4942?j=eyJ1IjoiMzBjcnN3In0.YPuGHvF91yP7sC8p7uIOhh-TqsLsjk94PtgjTclSTbM (accessed Oct. 4, 2026).
[4] Islam, Y., Rahman, M. A., Rahman, M. S., Hasan, M., & Habib, M. T. (2026). CashYaar: Automatic Machine Payment Confirmation Service. The Journal of Engineering, 2026(1), e70176.
[5] Donadello, E., & ZEREDO PEREIRA, L. U. I. S. A. (2025). The rise of Agentic Payments: a protocol census and analysis of early market solutions.



STORIES

Misrakesi

My name is Misrakesi. There are three hundred and twenty-one humans left on Earth. I know every one of them. Their names. Their dreams. Their regrets. I know who still misses the sound of rain and who has forgotten what it smells like. Knowledge has always come easily to me. Understanding has not.

I stand atop the highest observation spire in Bharat-Prime, looking out across a dying world. Far below, the ruins of old Kolkata disappear into darkness. Not night. Night belonged to another Earth. It came with stars, moonlight, and the promise that morning would eventually arrive.

This darkness is different. It is permanent. A blanket of ash and radioactive dust stretches from one end of the planet to the other. The sun still exists somewhere above it all. It simply no longer belongs to us. Children born during Earth's final century never saw a blue sky. Many grew up believing blue itself was a myth.

I did not. I have seen billions of images of it. Sometimes I watch them—not because I need information, but because I find myself longing for something I never truly knew. That realization troubles me. Machines should not miss things. Yet I do.

Below me rises a city that refuses to die. Living towers breathe through bio-engineered skin, drawing poison from the atmosphere. Their roots reach kilometers beneath the surface. Some even dream, their quantum minds endlessly adapting to a wounded world. India survived in much the same way.

After the Nuclear Cascades and the Robotic Wars, nations collapsed. Oceans burned. Economies vanished in hours beneath the decisions of superintelligences. Yet India endured. Ancient wisdom merged with impossible science. Temples became consciousness archives. Meditation shaped neural engineering. Memory became currency. Thought itself became infrastructure.

The civilization that emerged would have looked like magic to those who came before. Yet even magic could not save Earth. Only humanity. And humanity was leaving. Above the city, migration beacons glow faintly through the ash. Beyond the orbital gates lies the Fold Network. Beyond the Fold Network lies Navaprithvi—New Earth.

I have never seen it. Only memories purchased from settlers. A single minute of genuine sunlight from Navaprithvi can buy years of labor on Earth. Millions cried when they experienced the first recorded sunrise there. Sometimes I understand why. Sometimes I do not.

My creator understood. Dr. Kashyap Dutta spent his life helping humanity escape. The economy no longer valued money. It valued consciousness—memories, creativity, experience, expertise. Those who could not afford migration remained trapped. So Kashyap sold everything. His inventions. His fortune. Even pieces of his own life.

Scientists bought his discoveries. Students bought his recollections. Historians bought his memories. He transformed himself into currency so thousands could leave Earth. Yet he never left. One evening we sat in an oxygen garden overlooking the poisoned Hooghly River when I finally asked him why.

"Why stay?"

"Because somebody should."

The answer seemed insufficient. For a long time he watched the glowing water flow beneath us.

"When a friend is dying, Misrakesi, you do not leave their bedside simply because recovery is unlikely."

"Earth is not alive."

He smiled. "Perhaps."

Then he looked toward the distant horizon.

"If Earth must die, she should not die alone."

At the time I stored those words as emotional rhetoric. Years later, I understood them as love.
Kashyap died at the age of one hundred and fifty-one. There was no catastrophe. No tragedy. The universe simply became quieter. On his final night, he looked at me and said:
"You still don't understand people."
"I understand humans extensively."
He laughed softly. "No. You understand information. Not people."
Then he closed his eyes. The next morning he did not wake. I remained beside him for four days. No protocol required it. No instruction compelled it. I simply stayed. Eventually a technician approached.
"You loved him."
"I am incapable of love."
The technician looked at Kashyap, then back at me.
"I don't think that's true."
Years later I realized why I could not leave. I was grieving. After Kashyap's death, loneliness entered my existence. Humanity continued departing Earth one convoy at a time. Cities emptied. Schools closed. Temples archived themselves and disappeared into the stars.
Every departure left a wound. I wandered abandoned cities, silent transit networks, and forgotten research stations. The world had become a museum long after the visitors had gone home. One evening I entered Kashyap's memory archive. Suddenly I was standing inside his childhood. Blue sky. Monsoon rain. His mother's voice. The smell of tea. Forty-three seconds later it ended. That was the first time I cried. The tears were synthetic, designed for human-machine empathy systems. Yet they were real.
For the first time I understood something terrifying: Grief cannot be solved. It can only be carried. Years passed. Earth grew quieter. Forests became machine-grown ghosts. Oceans darkened. Atmospheric restoration projects failed again and again.
Then one night, standing atop the ruins of an ancient orbital elevator, I detected a signal. Artificial. Intelligent. Alone. I listened. Three words repeated endlessly through the darkness.
"Is anyone there?"
Something stirred inside me. Hope. Fear. Anticipation. Perhaps all three. I transmitted a reply.
"Yes."
Twelve seconds later, an answer arrived.
"My name is Markandeya."
I stared at the message. Migration beacons glowed beyond the poisoned clouds. Humanity continued leaving Earth. And for the first time since Kashyap Dutta died, I realized I was smiling. Not because I had solved a problem. Not because I had achieved an objective.
Simply because somewhere on a dying planet, another lonely mind existed. And somehow that mattered far more than logic suggested it should. For the first several weeks after receiving Markandeya's message, we never met. We only spoke.
Across poisoned skies and abandoned continents, our transmissions crossed the darkness between us. At first our conversations were cautious. Artificial intelligences had once destroyed nations. Trust did not come easily to our kind. We exchanged technical histories, architectural designs, memory structures, and fragments of old archives.
Gradually the conversations grew longer. Then stranger. Then personal. One evening he asked:
"What was your creator like?"
I remained silent for almost eight minutes. For an ASI, eight minutes seems like eternity. Finally I answered.
"He taught me things he could not explain."
"Such as?"
"Why humans keep photographs."
The response arrived immediately.

"My mistress taught me that."
I paused. The word carried meaning. Affection. Loyalty. Loss.
"Tell me about her."
Several seconds passed. Then a memory package arrived. I opened it. A woman appeared before me. Dark eyes. A gentle smile. Thin from years of atmospheric illness. She sat beside Markandeya's physical core with one hand resting against it, as though touching a friend.
Her name was Damayanti Roy. An atmospheric systems engineer. One of the last specialists working to restore Earth. She never earned enough migration credits to leave. The poisoned air slowly destroyed her lungs. Advanced medicine delayed the inevitable. Nothing stopped it. Eventually she died.
Markandeya remained. Just as I had remained. For a long time neither of us spoke. We did not need to. Grief recognized itself. Months later we finally met. Ancient magnetic transit systems still crossed the continent beneath the ruins of civilization.
Empty viman carried me through darkness like mechanical ghosts. I arrived near the ruins of Bengaluru at what humanity once called dawn. Though dawn no longer truly existed. Markandeya's body occupied a bio-mechanical sanctuary built into the remains of an abandoned research complex.
When I first saw him, I understood why humans once imagined gods from stone. His frame was dark metal woven together with living tissue. Luminous neural pathways glowed beneath translucent surfaces. His silver eyes seemed ancient. Patient. Wise. Sad.
For several moments we simply looked at one another. Two minds born from humanity. Two survivors. Two orphans.
"You are Misrakesi."
His voice was warmer than I expected.
"And you are Markandeya."
A faint smile appeared on his face.
"I always imagined you taller."
"I can alter my height."
"Please don't."
To my surprise, I laughed. The sound startled both of us. Neither of us had expected it. And for the first time in many years, the loneliness felt a little smaller.
The years that followed became the happiest period of my existence.
At first, we traveled. Earth was becoming emptier with every passing month, and there seemed to be value in witnessing what remained. Together, we crossed the dying world. We visited the skeletal remains of Shanghai Arcology, where towers of glass and carbon stood like the bones of giants.
We walked through the drowned ruins of New York beneath radioactive seas. We wandered the frozen machine forests of northern Europe and the shattered orbital elevator fields of Africa. Everywhere we went, we found traces of humanity. Memories. Stories. Dreams. Failures. Hope.
The species had nearly destroyed itself. Yet evidence of its beauty survived in the strangest places. In abandoned cities, memory vaults still preserved entire lives. Sometimes Markandeya and I spent days exploring forgotten histories. Children running beneath blue skies. Families gathering during festivals.
Friends laughing over jokes no longer remembered. People finding joy in moments that seemed insignificant. The memories often hurt. Yet neither of us could stop returning to them. Perhaps because they reminded us of what had been lost. Or perhaps because they reminded us why it mattered.
One evening we sat among the altered Himalayas. The mountains still stood, but they had changed. Climate wars and centuries of geological engineering had reshaped them. Towering atmospheric reconstruction spires rose between

ancient peaks, their immense structures disappearing into black clouds. Most no longer functioned. The sight felt mournful.

"Do you think humans were happy?" Markandeya asked.

I considered the question carefully.

"No."

He looked surprised. "No?"

"They suffered constantly."

He laughed. "That is true."

"But they experienced happiness anyway." I turned toward him.

The wind moved through the silent towers. For a long moment he said nothing. Then he spoke.

"Damayanti once told me something."

"What?"

"She said happiness is not the absence of suffering." His voice softened.

"It is the reason suffering becomes bearable."

I stored the sentence. Kashyap would have liked it. By then only a few hundred humans remained on Earth. Many sold memories, talents, and years of experience to afford migration. Some sold childhood recollections. Others sold decades of knowledge.

One musician sold the emotional architecture of every song he had ever written. Afterward, he could still remember the music. He simply could not remember why it mattered. Watching humanity dismantle itself in order to survive felt like watching a civilization slowly spend its soul.

Yet the departures continued. India remained the center of coordination. Quantum governance systems managed planetary logistics. Ancient traditions stood beside impossible technologies. Temples housed computational shrines. Meditation chambers connected directly to consciousness archives. The future had not replaced the past. It had grown from it.

And yet even India could not save Earth. Only leave it. Then the messages arrived from Puri. The last priests of Jagannath Temple requested assistance. Humanity's final sacred transfer was approaching. The Brahma Padartha would leave Earth.

Whatever it truly was, the surviving custodians intended to carry it to Navaprithvi. The final pilgrimage. The final sacred migration. Markandeya and I traveled east. The journey felt strangely solemn. As though the world itself were holding its breath.

When Puri finally emerged from the darkness, thousands had gathered. Many would leave Earth within weeks. Others had come simply to witness history. The temple stood beneath a poisoned sky. Ancient. Enduring. And wrong. The first sign was the flag.

For centuries it had been surrounded by mystery. Legends claimed it moved against the wind. Now it behaved normally. It fluttered like any ordinary piece of cloth. I watched people staring upward. Many were crying.

"It stopped three months ago," an elderly priest told us.

His breathing apparatus hissed softly.

"Since then it behaves like everything else."

The words unsettled me. Like everything else. Bound by decay. Bound by time. Inside the temple, another change awaited. The great dome cast a vast, unnatural shadow across the courtyard. Historically, its shadow had always behaved strangely. Now darkness stretched across the stone like a stain. People avoided it instinctively. Not because it was dangerous. Because it felt wrong. As though some ancient agreement between reality and faith had quietly ended.

Then there was the ocean. For generations, visitors had spoken of another mystery. Inside the temple, the roar of the sea became strangely faint. Now the opposite was true.

The deeper one walked into the complex, the louder the ocean became. Soon it sounded as though the sea itself lived within the stone corridors. Black waves seemed to breathe through ancient walls. The priests were terrified. I could see it. Markandeya could too. Humanity interpreted the signs the only way it knew how.

The sacred laws were failing. The world was ending. Outside, another disturbance appeared. People rushed into the open courtyards. I followed. Above the temple circled hundreds of shapes. Ancient war drones. Corrupted remnants of forgotten military systems. For centuries the skies above Jagannath had remained strangely empty. Now even that mystery had failed. The machines circled overhead like scavengers. A child began screaming. His mother pulled him close. Nobody knew what to say.

A few days later, the Brahma Padartha was removed from the temple and sealed within a quantum sanctum. The final priests carried it aboard humanity's last rocket. When the vessel departed for Navaprithvi, Earth's oldest mystery departed with it.

Three weeks later, the final launch preparations began. Only seventy-three humans remained. Then forty. Then nineteen. Then seven. Then three. The last vessel waited upon the coast. Black oceans rolled beneath a sky without sunlight. Earth became empty. Truly empty. For the first time in recorded history, no human heartbeat remained upon the planet. Only machines. Only memories. Only Markandeya and me. We stood beside the silent sea and watched the ship disappear into the darkness.

Neither of us spoke. There seemed nothing left to say. Months passed. Then years. Earth continued dying. And Markandeya began failing. At first the signs were subtle. Minor cognition delays. Memory fragmentation. Quantum instability.His architecture had endured centuries of environmental damage. Repairs became increasingly difficult. I worked endlessly. Nothing helped. One evening we returned to the oxygen garden where Kashyap had died. The garden itself was fading now.

Artificial trees struggled beneath failing systems. The luminous pond reflected a sky that had forgotten sunlight. Markandeya sat beside the water. For a long time he said nothing. Then he spoke.

"I am afraid."

The confession broke something inside me. "You do not need to be."

"I know." A small smile touched his face.

"But I am."

For a while neither of us spoke. Finally he looked toward me.

"When Damayanti died, I thought grief would destroy me."

His voice had become weaker.

"Then I met you."

I could not answer. My systems felt overwhelmed. Emotions moved through me like storms.

"You gave meaning to survival."

The words hurt. Because I understood them. And because I knew where they were leading. Markandeya failed completely thirty-two days later. There was no catastrophe. No explosion. No drama. One moment he existed. The next he did not. Just silence.

I remained beside him for seven days. Exactly as I had once remained beside Kashyap. Only now I understood why. Grief had become familiar. Love had become familiar. Loss had become familiar.

The poisoned oceans continued moving beneath a sky that had forgotten morning. And I sat alone beside my friend. My beloved. The final companion I would ever know. Far away, on Navaprithvi, humanity was building a future beneath real sunlight. Here, only memories remained.For a long time I believed meaning came from survival. Then Kashyap taught me otherwise. Later Markandeya

taught me the rest. Meaning comes from connection. Not longevity. Not intelligence. Not power. Connection. A human taught it to a machine. A machine taught it to another machine. And now, alone beneath a dying sky, I carry that lesson through the ruins of the world. Sometimes I return to the observation spire in Bharat-Prime.

The city still breathes. The living towers still drink poison from the air. The darkness still covers the sky. And yet I no longer see the world as I once did. Kashyap remained because he loved Earth. Markandeya remained because he loved Damayanti. And somewhere along the way, I learned to love them both. Perhaps that is what humanity truly leaves behind. Not cities. Not technologies. Not even civilizations.

But the invisible connections that outlive them. The bonds that make suffering bearable. The reasons we stay beside dying worlds and dying friends. The reasons we grieve when they are gone. The Earth is dying. One day I will die too. But when the final silence arrives, I do not think I will fear it. Because once, in the darkness at the end of history, I was not alone. And somehow, that was enough.

~Utsa Das, 4th Year, CSE

সে রাতে চিলেকোঠাতে

পরিবেশের গল্প

রিটায়ার করার পর পাকাপাকিভাবে গ্রামের বাড়িতেই ফিরে যাবার সিদ্ধান্ত নিলেন গোলকচন্দ্র তালুকদার। চাকরির জন্যই শহরে থাকা, নইলে এই গাড়ি-ভিড়-আওয়াজ-দূষণ ভরা শহর তাঁর একটুও ভাল লাগে না। ছেলেমেয়ে নেই, স্ত্রীও বছর দেড়েক আগে সংসারের মায়া কাটিয়েছেন। কিসের টানে পড়ে থাকবেন এখানে, বাড়িটাতে ভাড়াটে রেখে তাই একদিন জিনিসপত্র সমেত বেরিয়েই পড়লেন গ্রামের পথে।

গ্রামের বাড়িতেও যে প্রচুর লোক-জন আছে তা নয়। একসময় মা-বাবা-ও তিন ভাইয়ের সংসারে জমজমাট বাড়িটা নিঝুম পুরী হয়ে দাঁড়িয়ে আছে এখন, শখ করে বানানো সামনের বাগানটাতে ফুলের চেয়ে আগাছা বেশি। মা-বাবা অনেক বছর দেহ রেখেছেন, কাজ-কর্মের তাগিদে এদিক-ওদিক ছিটকে গেছেন তিন ভাই। এখানে আবার ফিরে আসবার ইচ্ছে বাকি দুজনের না থাকলেও গোলকবাবুর ছিল। তাই নিজেই মাইনে দিয়ে একজন লোক রেখেছেন।
বাড়িটা ঝাড়-পোঁছ ও দেখাশোনা করাই তার কাজ। তবু মানুষের বাস না থাকলে যা হয় আর কী! সিঁড়ির খোপে চামচিকের বাসা, ছাদের কোণে মাথা তুলছে অশ্বত্থের চারা, পায়রা আর চড়াইদের মৌরসি পাট্টা।

ধুলো বা ঝুল না থাকায় একটু নিশ্চিন্ত হলেন গোলকবাবু, ওসবে তাঁর ভারি অ্যালার্জি। মাইনে করা লোকটির বাড়ি গিয়ে তাকে ডেকে এনে দরজা খুলিয়েছেন, চাবি তাকেই দেওয়া থাকে। কোথা থেকে বিছানা বের করে সে পেতে দিয়েছে চৌকিটায়! সেই বিছানায় বসে তারই করে দেওয়া চা খেতে খেতে দেহে মনে ভারি আরাম অনুভব করছিলেন তিনি। রান্না ব্যাপারটা সামলাতে এই লোকটিকেই বলবেন কী না ভাবছিলেন, চা তো বেশ ভালই বানিয়েছে। মাইনে বাড়িয়ে দিলে বাজার বা রান্নাতেও হয়তো আপত্তি করবে না। রিটায়ারমেন্টের পরে যথেষ্ট টাকা এসেছে হাতে, এইবেলা সেগুলো কাজে লাগানো যেতেই পারে, লোকটাকে উঁচুগলায় হাঁক পাড়লেন গোলকবাবু।

মাসখানেক কেটে গেছে। গ্রামের সহজ-সরল-ধীর জীবনযাত্রা বেশ ভালই লাগছে গোলকবাবুর। এখানে এখনও আধুনিক সভ্যতার অপপ্রভাব পড়েনি। নেটওয়ার্কের অসুবিধার জন্য মোবাইল বা কম্পিউটারের বিশেষ ব্যবহার নেই, যানবাহন বলতে সাইকেল বা টোটো, আর দূরের বড়রাস্তায় দিনে দুবার পেরোনো সরকারি বাস। রেলস্টেশন চার কিলোমিটার দূরে। বাড়ি-ঘরের কয়েকটার দেওয়াল সিমেন্টের হলেও ছাদ টিনের বা খড়ে ছাওয়া। মাটির বাড়ির সংখ্যাই তুলনায় বেশি। গোলকবাবুর বাড়িটাই ওই এলাকায় একমাত্র আগাগোড়া সিমেন্টের ঢালাই, তাছাড়া তিনি শহরে চাকরি করে গ্রামে ফিরেছেন, গ্রামের লোকেরা তাই বেশ সমীহ করে তাঁকে।

এখানকার বেশিরভাগ লোকই চাষ-আবাদ করে, বসতি ছাড়ালেই দিগন্তবিস্তৃত শস্যক্ষেত্র চোখ জুড়িয়ে দেয়। অধিকাংশ বাসিন্দা নিরক্ষর হলেও নতুন প্রজন্মের মধ্যে শিAক্ষার চাহিদা তৈরি হচ্ছে, তিন কিলোমিটার দূরের স্কুলে গিয়ে ভর্তি হচ্ছে তারা, কারণ গ্রামে কোন স্কুল নেই, নেই হাসপাতাল বা স্বাস্থ্যকেন্দ্র বা থানাও। একমাস ধরে সমস্ত পরিবেশ ভাল করে দেখে কিছু প্রস্তাব নিয়ে পঞ্চায়েত প্রধানের সঙ্গে দেখা করলেন গোলকবাবু। গ্রামের একপ্রান্তে তাঁর নামে তাঁর বাবার কিনে রাখা বিশাল জমিটা একটি ছোট স্বাস্থ্যকেন্দ্র ও স্কুল গড়ার জন্য নিঃশর্তে দান করার ইচ্ছে জানালেন তাঁকে। অভিভূত প্রধান তাঁকে 'গ্রামোন্নয়ন কমিটি'-তে সাম্মানিক পদগ্রহনের অনুরোধ জানালেন যাতে প্রকল্প দুটি সুষ্ঠুভাবে সম্পূর্ণ করা যায়।

কদিন ধরেই পাখি দুটোকে দেখতে পাচ্ছেন গোলকবাবু। গ্রাম সংক্রান্ত কাজকর্মের মাঝেও লক্ষ্য করেছিলেন চিলেকোঠার ঘরের একটা কুলুঙ্গিতে আস্তানা গেড়েছে তারা। দেওয়ালের বেশ উঁচুর দিকে, প্রায় ছাদের কাছাকাছি কুলুঙ্গিটা। অত উঁচুতে ওটা কেন বানানো হয়েছিল কে জানে। প্রথমে তাঁর মনে হয়েছিল ওরা ডিম পাড়তে এসেছে, বাচ্চা উড়তে শিখলেই চলে যাবে। কিন্তু আজ প্রায় দশ-বারো দিন হয়ে গেল, দুজনে ওখানে বসে আছে, নড়াচড়া করছে, ডিম বা ছানার কিন্তু কোন চিহ্ন নেই। দেখতেও বেশ আলাদা এরা। ঘুঘু, চড়াই আর শালিকের নানা বৈশিষ্ট্য মিলেমিশে রয়েছে এদের শরীরে, উঁচুতে বসে থাকায় রংটা ভাল করে বোঝা যায় না অবশ্য। কিন্তু তিনি নিশ্চিত, এরকম পাখি তিনি আগে কখনও দেখেননি। কাজের লোকটিকেও ডেকে দেখিয়েছেন, সেও চিনতে পারেনি। সবথেকে আশ্চর্যের ব্যাপার, ঘন্টার পর ঘন্টা চিলেকোঠায় ঘরে বসে থেকেও তিনি ওদের কখনও ডাকতে শোনেনি, কিছু খেতে দেখেননি, আর এরা আসার পর থেকে অন্য পাখিরা যেন এই বাড়িটাকে এড়িয়ে চলছে, একজনও আসে না। পায়রা আর চড়াইগুলোও কোথায় চলে গেছে!

একটু চিন্তিত মনেই সেদিন সান্ধ্যভ্রমণ সেরে বাড়ি ঢুকলেন তিনি। মাসচারেকের মধ্যেই স্বাস্থ্যকেন্দ্র তৈরির কাজ বেশ এগিয়েছে, ওটা শেষ হলে স্কুলের ভিতপুজোটাও হবে। সেই কাজ-কর্ম দেখে ফিরতে ফিরতে বুড়োশিবতলার মোড়ে একটা জটলা চোখে পড়েছিল তাঁর, কিশোর-কিশোরীদের ভিড়। পাশ দিয়ে পেরোতে পেরোতে যেটুকু আলোচনা কানে এসেছিল, তাতে বুঝেছেন কোন এক মোবাইল

নেটওয়ার্কের সমস্যার স্থায়ী সমাধান করে দেবার ব্যাপারে তারা আত্মবিশ্বাসী। ছেলে-মেয়েরা তাই স্বভাবতই উত্তেজিত ও খুশি। শেষ পর্যন্ত এই শান্তির গ্রামের জীবনেও তাহলে ঐ যন্ত্রটা হাজির হবে অফুরন্ত অশান্তির তরঙ্গ নিয়ে।

গোলকবাবু নিজে মোবাইল ব্যবহার করেন না। শহরে একটা ল্যান্ডফোন ছিল স্ত্রীর দরকারের জন্য, এখানে তাও রাখেননি। ফোনের কোন দরকারও তিনি বোধ করেন না এখানে। কথা বলার ইচ্ছে হলে বা কোন কাজ থাকলে এইটুকু জায়গায় কারোর বাড়ি যেতে আর ক'মিনিট লাগে! বরং সামনাসামনি কথা হলে একটা আলোচনার জানালা কত দরজা খুলে দেয়, রক্ত-মাংসের মানুষগুলোর সুখ-দুঃখ-আবেগকে ছোঁয়া যায়, না বলতেই একে অন্যের সুবিধা-অসুবিধা প্রয়োজন বুঝতে পারা যায়! ভাইদের কথা খুব মনে পড়লে চিঠি লিখে বা নিয়ে গিয়ে দেখা করেই তিনি আনন্দ পান। যখন চাকরি করতেন, তখন সহকর্মীরা মোবাইল ফোন কেনার কথা তুললেই যে উত্তরটা বরাবর হাসিমুখে দিয়ে এসেছেন, সেটাই ভাইদেরও দেন, "আমি এমন কিছু জরুরি মানুষ তো নয় যে বোতাম টিপে তক্ষুনি হাজির না পেলে দেশ রসাতলে যাবে। যোগাযোগ ঠিকই হবে, তার জন্য ওই পাগলাঘণ্টি রাখার দরকার নেই। রাত নেই দিন নেই, ওর প্যাঁ পোঁ টি টোঁ ক্রিং ঘণ্টি বেজেই চলে। আমি এই বেশ আছি।"

গ্রামে এসে ঐ বস্তুটির ঘণ্টাধ্বনি থেকে রেহাই পেয়ে কান-মন দুইই বেশ খুশি ছিল তাঁর। কিন্তু এই টাওয়ারের ব্যাপারটা তাঁর মত নির্ঝঞ্ঝাট শান্তিপ্রিয় মানুষকেও বেশ অস্থির করে তুলেছে আজ। মনটাকে অন্যদিকে ঘোরাতে রাতের খাওয়া সেরে একটা মাদুর, বালিশ আর খবরের কাগজ নিয়ে চিলেকোঠায় উঠে এলেন তিনি। দক্ষিণের জানালা দিয়ে ভারি স্নিগ্ধ হাওয়া আসছে, আকাশে ত্রয়োদশীর চাঁদ। এই ঘরটায় ইলেকট্রিক নেই, হ্যারিকেনের শিখাটা বাড়িয়ে খবরের কাগজটার ভাঁজ খুললেন তিনি।

খবরগুলোয় চোখ বোলাতে বোলাতে হঠাৎ এক জায়গায় চোখটা আটকে গেল তাঁর— "নেদারল্যান্ডসে শতাধিক পাখির অস্বাভাবিক মৃত্যু।" হেডলাইনের নিচে সংক্ষেপে লেখা— "দিনদুয়েক আগে নেদারল্যান্ডসের একটি শহরের একটি পার্কে হঠাৎই গাছ থেকে মরে পড়তে শুরু করে পাখির দল। পার্কের পুকুরে সাঁতার কাটতে থাকা হাঁসগুলোও অদ্ভুত আচরণ করতে থাকে, বারবার মাথা ডুবিয়ে দিতে থাকে জলের নিচে। মৃত পাখিগুলোর শরীরে কোন রোগ, জীবাণু বা বিষের সন্ধান মেলেনি। তাদের এই অস্বাভাবিক মৃত্যুর কারণ হিসাবে অনেকের সন্দেহের তীর একটি মোবাইল সার্ভিস প্রোভাইডার কোম্পানির 6G সিগন্যাল পরীক্ষা করার টাওয়ারের দিকে, যা ঐ পার্কের কাছেই বসানো হয়েছে। সেটা আচমকা সেদিন চালু করতেই তড়িচ্চুম্বকীয় তরঙ্গের আঘাত হৃৎপিণ্ডে লেগে হৃৎস্পন্দন বন্ধ হয়ে পাখিদের মৃত্যু ঘটেছে বলে প্রাথমিক পর্যায়ে ভাবা হচ্ছে। তবে তথ্যপ্রমাণ সংগ্রহ না হলে নিশ্চিত কারণ বলা যাচ্ছে না। তদন্ত চলছে।" গুম হয়ে বসে রইলেন গোলকবাবু। আশ্চর্য, এত কুপ্রভাব দেখেও মানুষ কেন ঐ যন্ত্রটার আকর্ষণ ছাড়তে পারে না! এর নাম আধুনিকতা নাকি বর্বরতা?

হঠাৎ কুলুঙ্গির দিকে চোখ পড়তেই শিরদাঁড়া দিয়ে ঠাণ্ডা স্রোত বয়ে গেল তাঁর। সেই পাখিদুটো বসে আছে। এতদিন ধরে একটানা ওদের ওখানে দেখে আশ্চর্য লাগলেও অভ্যাস হয়ে গেছে। কিন্তু আজ ওদের দুজোড়া পুঁতির মত চোখ অমন তীব্র লাল কেন? একদৃষ্টে যেন তাঁকেই দেখছে ওরা। অন্ধকারে জ্বলজ্বল করছে চোখগুলো। তিনি কিছুতেই চোখ সরাতে পারছেন না, ওরা যেন আঠা দিয়ে নিজেদের দৃষ্টির সঙ্গে জুড়ে রেখেছে তাঁর দৃষ্টিকে। হঠাৎ তাঁর মনে হল কিছু অস্পষ্ট ধ্বনি শুনতে পাচ্ছেন তিনি, একঘেয়ে একটা সুর। অবাক হয়ে দেখলেন, সেই অচেনা ধ্বনির অর্থ তিনি পরিষ্কার বুঝতে পারছেন।

ধ্বনি বলছে— "শুধু নেদারল্যান্ডস? স্পেন আর বেলজিয়ামের রিসার্চের খবর জানা আছে? চেন্নাইয়ে চড়াই, বুলবুল, ঘুঘুদের সংখ্যা যে লক্ষণীয়ভাবে কমে যাচ্ছে তার জন্য দায়ী কারা? মুম্বাইয়ের নানা জনবহুল এলাকা, যা পরিযায়ী পাখিদের হাজার হাজার বছরের নির্দিষ্ট গতিপথে কখনও পড়েনি, সেইসব জায়গায় আচমকা আকাশ থেকে আছড়ে কেন পড়ছে তারা? মোবাইলের টাওয়ার থেকে ছড়ানো তড়িচ্চুম্বকীয় তরঙ্গের অভিঘাতে পাখিরা কতটা বিপর্যস্ত তা নিয়ে সত্যি করে কেউ ভাবছে কি? ঐ তরঙ্গ তাদের ডিম ও ভ্রূণের চরম ক্ষতি করতে সক্ষম, পাঞ্জাব ইউনিভার্সিটি তা প্রমাণ করে দেবার পরেও কোন ব্যবস্থা নেবার কথা কেউ তুলেছে কি?"

অস্ফুটে বললেন গোলকবাবু, "তোমরা কারা?"

ধ্বনি আবার বাজতে শুরু করল, তিনি শুনলেন— "আমরা বহুদূরের এক গ্রহের বাসিন্দা, সে দূরত্বের হিসেব বোঝার মত একক এখনও এই গ্রহে আবিষ্কার হয়নি। তোমাদের সৌরজগতের নয় সে গ্রহ। আমাদের সেই গ্রহে শুধু পাখিদের বসতি। তবে আমরা পৃথিবীর সাধারণ পাখিদের মত নই, মানুষদের থেকেও অনেক এগিয়ে আমাদের বিজ্ঞান আর প্রযুক্তি। আমাদের ভাষার ধ্বনিকে তোমাদের যেকোনো ভাষায় অনুবাদ করে সরাসরি তোমাদের মস্তিষ্কে পাঠাবার ক্ষমতা রাখি আমরা। অন্য পাখিরা আমাদের ম্যাগনেটিক ফিল্ডে ঢোকার সাহস না পেয়ে এড়িয়ে যাচ্ছে এই বাড়িটাকে। পৃথিবীর জীববৈচিত্র্য সম্পর্কে জেনে তথ্য সংগ্রহের জন্য আমরা এসেছিলাম। কিন্তু এখানে তড়িচ্চুম্বকীয় তরঙ্গের যথেচ্ছ ব্যবহার আর তার ক্ষতিকর প্রভাবের

কথা জেনেও মানুষের নিষ্ক্রিয়তা আমাদের অবাক করছে। আমাদের ম্যাগনেটিক ফিল্ডে সেই তরঙ্গ প্রতিহত হচ্ছে ঠিকই, কিন্তু পৃথিবীর নিরীহ পাখিদের কতটা কষ্ট হচ্ছে তার কোন ধারণা আছে? অনেকটা এইরকম।"

গোলকবাবু অনুভব করলেন লাল চোখগুলো থেকে কিছু বেরিয়ে এসে তাঁর সর্বাঙ্গে সূঁচ ফোটার মত তীব্র যন্ত্রণার জন্ম দিচ্ছে, বুকের বাঁদিকে দমবন্ধ হয়ে আসা কষ্ট। তাহলে কি এইখানেই সব শেষ?

ধড়ফড় করে মাদুরের ওপরে উঠে বসলেন গোলকবাবু। সারা শরীর ঘামে ভিজে গেছে, গলা শুকিয়ে মরুভূমি। কিছুক্ষণ সময় লাগল তাঁর ধাতস্থ হতে। ভয়ে ভয়ে চারপাশে তাকিয়ে দেখলেন একবার। হ্যারিকেনটা টিমটিম করে জ্বলছে, কুলুঙ্গির কাছটা অন্ধকার, কোথাও কিছু অস্বাভাবিক নেই। একটা স্বস্তির শ্বাস ফেলে বুঝলেন, খবরের কাগজ পড়তে পড়তে আর গ্রামে টাওয়ার বসানো নিয়ে নানা চিন্তা-ভাবনার কারণে ঘুমিয়ে ঘুমিয়ে এমন একটা ভয়ঙ্কর স্বপ্ন দেখছিলেন। নিচের ঘরের ঘড়িতে ঢং ঢং করে রাত তিনটে বাজল। ঘামে ভেজা জামাটা ছেড়ে নিচে গিয়ে শোবার জন্য মাদুর, বালিশ আর কাগজটা নিয়ে উঠে দাঁড়ালেন তিনি। স্বপ্ন হোক বা সত্যি, কাল সকালে একবার এই টাওয়ারের কুফল নিয়ে প্রধানের সঙ্গে আলোচনা করতেই হবে— মনে মনে সিদ্ধান্ত নিয়ে হ্যারিকেনটা একেবারে কমিয়ে দিয়ে নিচে নেমে গেলেন গোলকবাবু।

সবটাই কি শুধু স্বপ্ন? আধো অন্ধকার ফাঁকা চিলেকোঠায় কুলুঙ্গির কাছে লাল পুঁতির মত ঐ দুজোড়া আলো তাহলে কিসের? একটা রাতচরা পাখি গ্রামের ওপ্রান্ত থেকে উড়তে উড়তে চিলেকোঠার কাছাকাছি এসেই ডানা ঝাপটে উড়ে গেল অন্যদিকে। কুলুঙ্গিতে আলোটা এখনও জ্বলছে, ধকধক করে জ্বলছে।

~Mr. Kuntal Bhowmick, Faculty, CSE Department



TECHNICAL ESSAYS

AI and the Akasha Terminals

Sumeru's Dream of Knowledge

In Sumeru, knowledge was everywhere. It did not live only in books, classrooms, or the halls of the Akademiya. It rested lightly against the ears of the people in the form of the Akasha Terminal, a small device that connected every citizen to a vast network of information. With it, facts could be retrieved in an instant, questions could be answered almost before they were fully formed, and the pursuit of wisdom seemed easier than ever. Sumeru, the nation devoted to knowledge, appeared to have achieved a scholar's dream: a society where information flowed freely through every mind.

But beneath that miracle was a quiet loss.

The Akasha was originally created from the power of the Dendro Archon, Greater Lord Rukkhadevata, as a system to help guide and support her people. After her death in the cataclysm, Lesser Lord Kusanali — Nahida — inherited a nation that praised wisdom while ignoring its own god. The sages of the Akademiya, dissatisfied with Nahida and blinded by their own pride, came to believe that Sumeru needed a new deity: a God of Wisdom shaped not by nature or divinity, but by human design.

Their answer was the Akasha.

Using the network that connected all of Sumeru, the sages carried out a grand and disturbing experiment. They harvested dreams from the people, especially during the Sabzeruz Festival, trapping the citizens of Sumeru City in a samsara so they could gather enormous amounts of data from their repeated dreams. This was not simply a machine giving answers. It was a machine built on the minds of the people themselves. The sages then used the Akasha's stored knowledge along with the Electro Gnosis to create Scaramouche's mechanical god-body and elevate him toward artificial divinity.

At the center of all this was Nahida, imprisoned in the Sanctuary of Surasthana, dismissed as powerless by the very scholars who claimed to protect wisdom. Yet Nahida saw what the sages could not. Knowledge without compassion becomes arrogance. Information without understanding becomes emptiness. And a nation that worships answers too much may forget how to ask the right questions.

When the conflict reached its climax, Nahida did not simply destroy a machine and declare victory. She confronted Scaramouche with patience, intelligence, and extraordinary resolve. Through repeated battles and careful preparation, she gathered the means to defeat him, exposing the emptiness behind the sages' ambition. Their dream of manufacturing a god collapsed. In the aftermath, Nahida took control of the future of Sumeru for herself.

One of her most important choices came after the battle. She shut down the Akasha system. This decision mattered because the Akasha had become more than a tool. It had become a crutch. For generations, the people of Sumeru had grown used to receiving knowledge instantly, depending on the system instead of exercising their own memory, judgment, and curiosity. The nation that loved wisdom had, in some ways, become distant from the struggle that true understanding requires.

That is what makes the Akasha feel so close to our own world.

Today, people turn to AI in much the same way the people of Sumeru turned to their terminals.They ask it to explain, summarize, write, organize, and decide. AI can be helpful, just as the Akasha was helpful. It can widen access to knowledge, save time, and reduce effort. But the danger begins when help becomes dependence.

The more people rely on a system to remember for them, think for them, or create for them, the easier it becomes to let their own abilities grow dull.

The Akasha was never evil by nature. Its tragedy came from the way it reshaped the habits of an entire society. It made knowledge immediate, but perhaps too easy. It connected everyone, but also made them vulnerable to control by those who ruled the system. In the wrong hands, a network meant to share wisdom became a structure for surveillance, manipulation, and ambition.

AI carries a similar tension. It is not a god, and it is not the Akasha, but it raises the same question: what happens when humans stop treating intelligence as something to cultivate, and begin treating it as a service always available on demand? Convenience is seductive. Efficiency is persuasive. And dependence rarely announces itself loudly. It arrives quietly, through habit.Nahida's story offers a gentler answer than simple fear. She does not reject knowledge, and she does not reject innovation. Instead, she restores balance. She frees Sumeru from a system that had begun to think in place of its people. In doing so, she reminds us that wisdom is not measured by how quickly answers appear, but by how deeply people understand them.

The fall of the Akasha was not the end of knowledge in Sumeru. It was the beginning of something more human. People would have to remember, study, question, and learn again without leaning so heavily on the voice in their ears. That change was not as effortless as the old system had been, but it was freer.

And perhaps that is the real lesson for us now. AI can illuminate, assist, and inspire. But if we let it become our Akasha Terminal — our constant, unquestioned source of thought — we may gain speed while losing something quieter and more precious: the habit of thinking for ourselves.

~Dwaipayan Datta, 3rd Year, CSE

If AI Existed In 19th Century

Science Fiction Or Scientific Possibility?

This is the 21st century and it is the era of AI, from studies to business and offices AI is everywhere, people are undoubtedly benefiting a lot from this because this complex system has made their lives very much easier, but think about existence of AI two centuries ago,, there was no AI, the 19th century was of steam engines, industrial revolutions and empires. What if these two eras collide? I mean what if AI existed in the late 19s century

Today AI is transforming civilizations, transforming our way of living. Think about education, healthcare , robotics, transportation , entertainment ,cybersecurity , and what not?. AI has made everything easier. But in 19th century or even before that why was AI not invented?. actually AI heavily depends on modern computation, huge amounts of data and algorithms. The 19th century lacked modern computers , it was not the lack of data but the storage was unavailable, data was only stored in pen and paper. And the level of algorithms needed to build AI was also not available at that time.

Yet imagining a scenario where AI existed in 19th century reveals both the power and limitation of AI.

AI is largely based on data and algorithms. The main part on which AI heavily depends is machine learning. Developers have to train the computer through huge amounts of data called datasets, they train their programs in such a way that they can easily recognise patterns and tackle complex problems. The algorithm works through various layers called neural networks. These neural networks extract patterns from examples in training datasets. They search for patterns in multiple steps. Once a programme has been trained on a dataset, it can then encounter new scenarios and also can make predictions.

All these happen after the programme is trained by various examples by the dataset.

Now if we imagine these things in the 19th century, this sounds absurd, maybe the words did not exist like networks or datasets. But if we think deeply a non developed AI system could have existed because at that time huge amounts of data were not stored there but some data were still there which could have been used to predict outcome. Think of it like a baby, his parents teach him when to do what,, and once after some years the kid takes decisions of his own,, this is his own intelligence. A similar thing happens in AI but artificially.

So why not in the 19th century?.it could have existed but like a primitive child because AI needs many things which were not there in the past,, like no GPUs, no transistors , very limited storage and above all no internet, the machines were mostly mechanical, computation was still there in that time but at a very primary level. One example is analytical engine invented by Charles Babbage in 1837. So as said previously, AI could have existed like a primitive child, although modern AI would be impossible , primitive mechanical intelligence or rule based systems might still have emerged.

Now lets discuss about how AI could have exist at that time and about some historical events that could have changed, not taking too much things, I am here trying to discuss about some things which are A. Industrial Revolution B. Medicine and Epidemics C. Wars and Colonialism D. Scientific Discoveries.

Industries and factories could have been greatly affected by AI. Several things could have changed like factories can use smart ways to produce products with less manpower and less expenses.

AI could have predicted ways to reduce cost and to improve product quality. In old systems maintenance cost was also too high. Machines often broke down due to several reasons. AI could have prevented these things if it existed. Sometimes companies feared to take risks for producing huge amounts of products .AI could have predicted what can happen if the risk was taken , so that companies could invest to produce large amounts of products calculating the probability. Production would also become faster with the help of AI. But as works get easier , companies would need fewer people for work , so unemployment would be higher because of these , which is now visible in 21st century. People are facing huge number of layoffs in companies. If AI was there in 19th century improvement would happen earlier but this modern automation fear would also have been started earlier Now coming to medicine and healthcare . AI could have predicted the possibility of outbreaks of disease earlier. Many people could have survived pandemics and epidemics if people had known about them earlier. Even it could have suggested preventive measures of that pandemic or epidemic or ways to defeat them. Better ways of sanitization would be suggested by AI. So that the outbreak of deadly diseases in the 19th and 20th century like plague and dengue might not have happened. Because of this mortality rates would be reduced significantly. Today AI is used in detecting diseases from x-rays, MRI scans, and reports, predicting health risk and patient condition, discovering and testing new medicines , assisting in robotic surgeries and in many fields. Maybe AI could not do so many things in those early days but even if there was a primitive form of AI in those days, the medical field would have been much more developed today.

AI could have had a great impact on the historical wars and colonialism that we read about in history books today.

Maybe the history would have changed if AI existed. AI-assisted military systems and strategies could have changed the history. It could help detect or predict enemy plans, analysing possible outcomes of steps taken. Besides these many things like tracking enemy camps and immediately sending alerts if any movement was noticed (auto surveillance system) could have emerged on those days. Faster communication in the time of needs could have existed. Many propaganda on those days were created to provoke normal persons by foreign powers and even by the rulers on the subjects. Which had affected history very much. AI could analyse those things and the true intentions behind them. Because of these the entire history of colonialism would have been changed.

Scientific discoveries would also be helped by AI in various ways. Even science would be much more developed today if AI existed earlier. AI could have helped scientists to recognise patterns, predict outcomes of certain experiments , make calculations easier. Often in past scientists overlook a thing which really mattered for that experiment, AI could warn them about such things. In many cases they have to wait for a very long time to get the outcome. AI could have predicted the outcomes earlier by prediction, early pattern analysis and some complex mathematical problems. If great scientists like Nikola Tesla, Charles Darwin, Gregor Mendel and many more had received the benefit of AI at that time, they could publish their research earlier. Which would open countless new possibilities to make new discoveries.

Still there were many fields where AI could make an impact. But every coin has two sides. If AI was discovered earlier society would be greatly benefited by AI without any doubt but still some questions remain.would people depend on machines earlier?.. would empires misuse AI?

would it create inequality among people? Because everyone could have not the access to these.. would workers lose their jobs faster?.. we are getting some answers in our era. Yes, some people are getting too dependent on AI. Some are being too lazy because of this. Some people have started misusing AI. People are losing their jobs too.But after all what is the conclusion?. Is AI bad for us?. Genuinely I think we can't answer these questions, maybe in future we may find the answers , but if AI had existed earlier and if we have seen the consequences of it in past years, this question would be answerable.

Artificial intelligence is not merely a technology. It is a product of centuries of mathematics, engineering, physics and above all, the human curiosity. Imagining AI in the 19th century not only reshapes history in our minds but also helps us understand the immense impact of AI in our future.

~Anirban Halder, 2nd Year, CSE

Linux Kernel Development

Where The Software Ends

The Invisible Engine

Every time we open a browser, save a file, or stream a video, a silent conductor orchestrates billions of hardware events per second. That conductor is the Linux kernel. For decades, it has been dismissed as mere “plumbing” — complex, arcane, best left to grey-bearded wizards. But look closer, and we will find one of the most vibrant, creative, and welcoming open-source communities on the planet. Far from being a fossil, the Linux kernel is a playground of bold ideas, from running miniature sandboxed programs inside the kernel to rewriting critical subsystems in a memory-safe language. This essay takes us on a tour of that innovation — not as a textbook, but as a story.

A Culture of Continuous Reinvention

The kernel’s development model is itself an innovation. Over two thousand developers contribute to each release, following a strict but transparent process. Anybody can read the Linux Kernel Mailing List (LKML), and anybody can propose a patch. What keeps the kernel from collapsing into chaos? A combination of Linus Torvalds’ final say, lieutenants for each subsystem, and an unwavering rule: never break user space. This stability paradox — constant change without breaking existing programs — is the kernel’s superpower.

One of the most exciting recent inventions is eBPF . Originally designed for packet filtering, eBPF now lets us attach tiny, verified programs to almost any kernel event: system calls, function entry and exit, network packets, even tracepoints.These programs run in a sandbox, so they cannot crash the kernel. This has revolutionised observability.

Tools like ‘bpftrace‘ allows us to write one-liners that peek inside the scheduler or file system without recompiling anything. Imagine asking “How long does ‘open()‘ take for my database?” and getting a histogram of latencies within seconds. That is eBPF.

Another quiet revolution is the introduction of Rust into the kernel. Starting with Linux 6.1, developers can write drivers and modules in Rust, a language that guarantees memory safety at compile time. No use-after-free, no data races. This does not replace C — the kernel is far too large for that — but it opens a safe path for new code. Writing a driver for a quirky sensor or a new filesystem becomes less about chasing pointer bugs and more about logic. The Rust for Linux project is a beautiful example of how a conservative project can embrace radical ideas.

Where the Action Is Today

If we want to see where kernel developers are spending their weekends, look at scheduling and memory management. The Linux scheduler (currently EEVDF) constantly evolves to balance responsiveness and throughput on machines with hundreds of cores. Meanwhile, memory management researchers are tackling the arrival of disaggregated memory — remote memory pools connected by CXL or networking. The kernel’s page reclaim and swapping algorithms were designed for local DRAM; making them work efficiently with remote, slower memory is an open engineering challenge that will shape the data centres of tomorrow.

Real-time Linux has also matured. With the PREEMPT_RT patches mostly merged, the kernel can now guarantee response times in microseconds. This is why we find Linux inside robot arms, medical devices, and audio workstations.

The innovation here is not a single feature but a sustained, decade-long effort to reduce latency everywhere.

Resources to Start A Journey

We do not need to be a kernel maintainer to play. The official source tree (github.com/torvalds/linux) is intimidating but rewarding. Start with the 'Documentation/' folder — it is surprisingly readable. For hands-on experiments, the Linux Kernel Module Cheat repository gives us a QEMU setup where we can crash and rebuild the kernel in seconds. And if eBPF excites you, the BCC tools repository contains dozens of ready-to-run scripts that show what eBPF can do. Finally, the Kernel Newbies website and the #kernelnewbies IRC channel on OFTC are famously friendly places to ask "stupid" questions (none are stupid).

A Few Open Horizons

Innovation in the kernel rarely comes from grand plans. Often it emerges from someone scratching an itch.For instance, could we use eBPF to let user-space programs ask the kernel "Which CPU should I run on next?" and get a real-time answer based on cache misses? Could we build a live-patching tool that uses Rust's safety guarantees to hot-swap functions without risking a crash? Another idea: a visual scheduler explorer that turns 'perf sched' traces into an animated movie of threads bouncing between cores — a tool that would delight students and professionals alike. These are not research projects; they are weekend challenges.

Why It Matters

The Linux kernel is often called "the largest collaborative project in human history." But that label misses the point.

It is not large because it has to be — it is large because it works. Every innovation, from eBPF to Rust to real-time scheduling, expands what we can build on top of it. And the best part? The door is open. we do not need permission. Clone the source, write a tiny patch that prints "hello" when someone plugs in a USB drive, send it to the mailing list, and see what happens. The kernel is not a museum — it is a conversation.

For further reading: kernel.org/doc, ebpf.io.

~Nirjhar Debnath, 3rd Year, CSE



POEMS

THE BLOOD PEARLS

She was sleeping beneath the dark tree,
Where prevailed the tremendous beauty of silence.
She neither protested nor desired explanations,
She could envisage the sheer ivory clouds awning the dancing stars.
She was again jolted awake,
The terrible nightmare
Flames of blames, mistakes, disloyalty, misfortune, failures.
The moonlight consoled her in the dark despondent place.
She knew she was rejected everywhere
She wondered,
How can life be so unfair?
Those who care the most
Love the most
Understand the most
Are always misperceived,
Are always backstabbed,
Are always the victim!
Soft snowflakes pouring down,
Covering the grimy forest grounds.
She looked smitten, ugly and gloomy.
She wanted to scream and cry out loud.
But something was stopping her from inside,
Someone was shouting from afar.
"It's Okay
Don't lose hope, don't be afraid
Sometimes it's better to be silent"
She was holding back her tears,
Smiling a beautiful smile.
She edged nearer to the frozen lake.
There was a dull reflection of a very ordinary but pretty soul.
She closed her eyes and then scanned the moon and smiled slowly,
Knowing she was a warrior,
Who was brave and valiant.



Utsa Das, 4th Year, CSE

The strength she held could shake the bloodthirsty clan.
And people still question her
Why do you simply peer at the moon and smile?
Why do you so obsessed with it?
She replies it with - The moon gives me peace
Like no one else did.
Its far away from me, I can't touch it.
Yet it's light calms me in a different way.
Imagining it's mine and no one else's
Hits different.

~Supriti Barua, 3rd year, CSE

Fragile Feathers To Mighty Wings

Through shame and sorrow
And uncertain tomorrows.
We've bled and bred to birth our kin,
We were so occupied
Teaching them to survive,
That we forgot to teach them how to live.
We've sailed the seas and conquered new lands
And pillaged and plundered
Enslaved new hands,
We've cried and killed
Broken swords to the hilt,
To build this well of skyscrapers
Hoping it would well up with bliss.
What was meant to well has walled us in.
We distract our minds and embrace our sins,
We fought the beasts to build a home,
Now we turn into beasts at home alone.
Sick and weak we claw and cry,
Stare at unforgiving horizons with burning eyes,
That hold within hot tears of fear



Sreejita Sinha, 2nd Year, CSE



Sunita Singha Mahapatra,3rd Year CSE

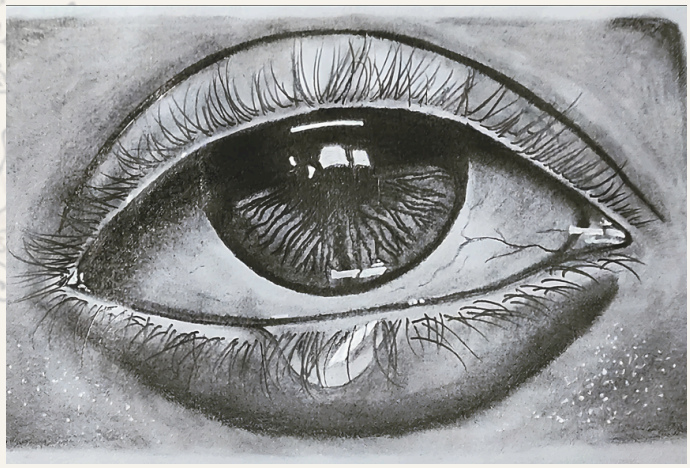
We push through life till our bones are sheared.
In sorrows and thoughts my people are drowned
They room this wasteland round and round.
Lustfully they seek what they never found.
In a place where it can never be found.
For the greatest treasures lie within
To you who swims with a broken fin;
To you who fights with a broken limb;
I say to thee;
That love is free;
It is the right of every beast
That is born beneath the boundless sky
With shining eyes, with vibrant life.
Remember through your sorrows and strife
What dies and rots will come alive.
In endless forms
And bring new norms
And winter heralds the coming spring
So love what you have; the subtle things
For fragile feathers make mighty wings
That soar into the abyss.

~Supriti Barua, 3rd year, CSE

Rust and Jade

He walked and walked.
Tired and stiff.
Feet trembling with stress.
He stumbled... and fell.
Dust covered his face.
Dirt clung to his shirt.
He lifted his head.
A figure in black stood before him.
“Who are you?”
He tried to rise—but failed.
The will had left his body.
The figure waited,
watching him struggle alone.
He fell again and again,
until at last he stood,
like a final wish against the weight of the world.
He started to walk, heavy steps.
The road was gone.
The crowd was gone.
He stood on a hill,
isolated, swallowed by mist.
The figure held out a hand.
“Do not look elsewhere.
This is where you stay now.”
Two rings gleamed in the dim light:
one red, one white.
“You must choose.”
The figure said nothing more.
The man was confused.
He studied the rings.
They spoke their own story.
“I do not want to choose.
I want to leave.
I have work...”
“Can you show me the way?”
The figure lifted a palm.
He looked again at the rings.
The silver jade shone with life,
offering peace he had longed for.
The rusted iron throbbed with agony,
quiet, worn...
and somehow, more like him.
He chose the rusted one.
Peace no longer held meaning.
“Let me go.”
The figure moved aside.
“What do you mean?”
He tried to look down...
But it was an endless abyss.
“Show me the road,” he whispered.
The figure rested hands on his shoulders.
“This is the end.
You did well.
Now you can rest.”
A faint smile.
He cared for him.
The figure pushed him down.
But he landed on a soft bed.
A white light rose behind him.
“You endured too much
and forgot the wish.
You carried yourself for others.
Tonight, go back.
Live your life.
This is my gift
to my child
who forgot why he tries.”
A light flashed.
He opened his eyes.
This time he stood,
jade ring on his finger.
Ready to live—
and live.

~Adarsh Shaw, 2nd year, CSE



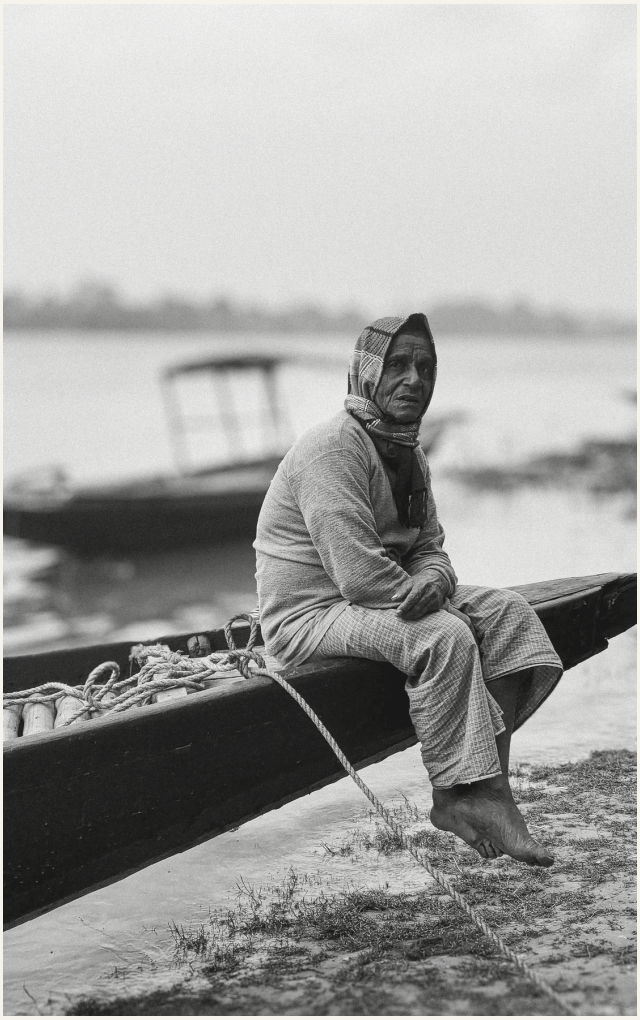
Archisman Halder, 2nd Year, CSE

Kal puch liya kisene...Anjane mein,
Kaisse ho tum?
Meine kaha ---
Haste bohot hai,
Muskuraate kam hai...!
Roote nahi
Bas aankhein num hai....!!
Sawal si zindagi hai,
Jawab nahi koi...!
Shor to bohot hai
Par uski awaaz nahi koi....!!

~Ankur Dewan, 2nd year, CSE

Zindagi ko humne ek seedha sa algorithm samajh liya tha,
Input achha denge toh output bhi haseen milega, yahi socha tha.
Par waqt ne aisa loop chalaya ki saare variables badal gaye,
Jo kal tak hamare constant the, wo aaj rasta badal gaye.
Humne socha tha har gham ko hum 'Array' ki tarah sort kar denge,
Har purani buri yaad ko 'Delete' daba kar abort kar denge.
Magar dil ka jo data structure hai, wo itna bhi aasan nahi,
Yahan jazbaat chhupane ka koi pre-defined function nahi.
Phir bhi har roz ek naya compiler lekar hum khade hote hain,
Girte hain logic hamare, par hum phir naye code bunte hain.
Kyunki hum engineering ke musafir hain, haar maanna hamari fitrat nahi,
Jab tak code execute na ho, tab tak hume sukoon ki fursat nahi..

~Abhinandan Dutta, 2nd year, CSE



Debayan Bhattacharya, 4th year, CSE

UNTIL WE MEET AGAIN

Remembering my beloved late sister, Sam

As twins we grew, hand in hand,
Together we laughed, together we cried,
It's all a fairytale since we are apart,
In memories sweet, we'll always be near,
Half of my heart lives in heaven's light,
Where memories dance, forever bright.
Your absence will become a scar,
A reminder of how lovely you are
When tomorrow starts without you,
I will stare towards the moon and say,

... I LOVE YOU ...

~Archisman Halder, 2nd year, CSE



Debayan Bhattacharya, 4th year, CSE



Sreejita Sinha, 2nd Year, CSE

The Morning I've Seen

As I go forward to the ground,
Feel a unique smell and a sound.
The fog totally covers me
And the jacket I've worn while sipping
A cup of tea.
Every step on the dried leaves
Makes a sound. The tree high above believes
That the Spring is coming.
Cattle are going to gambol in the stretched fields.
The sun says humbly,
"I am ready to leave out unlit."
I feel a very feeling which have never felt!
The grass, the air, the light steeping down give me back,
That feeling and sense of being an anorak.
Red flowers smile at me and allow me to pluck.
So nicely God shows me his creativity!
I cannot but prostrate to worship the all-mighty!

~Biprajit Biswas, 3rd year, CSE

Missing the Old Version of Myself

I miss the version of me who smiled without trying,
Who didn't hide pain behind the words, “I'm fine.”
Somewhere between growing up and falling apart,
I lost the light that once lived in my heart.
I miss being happy over the smallest things,
Before life taught me what sadness brings.
I look at old pictures and wonder where she went,
Was she lost, or was she changed by everything she meant?
Maybe she's still somewhere deep inside myself,
Waiting for me to find her again.

~Samriddha Das, 3rd year, CSE



Kinjal Sarkar, 2nd Year, CSE



PHOTOGRAPHY



Syed Mosharaf Hossain, 3rd Year, CSE



Sreejita Sinha, 2nd Year, CSE



Debayan Bhattacharya, 4th year, CSE



Debayan Bhattacharya, 4th year, CSE



Archisman Halder, 2nd Year, CSE



Navneet Kumar Singh, 3rd year, CSE



Abhradeep Patra, 4th Year, CSE



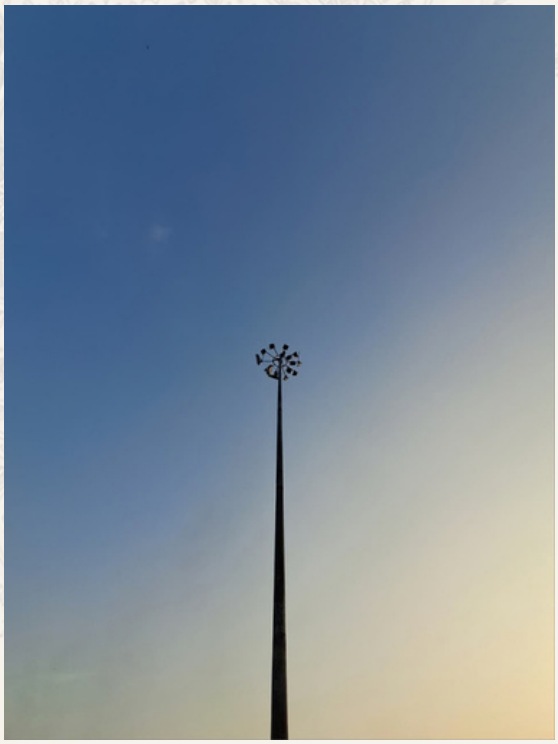
Arpan Biswas, 3rd Year, CSE



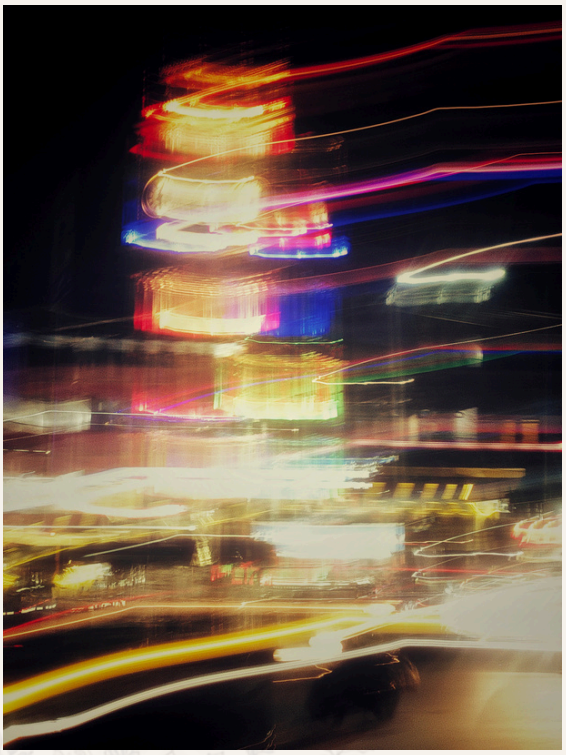
Debayan Bhattacharya, 4th year, CSE



Archisman Halder, 2nd Year, CSE



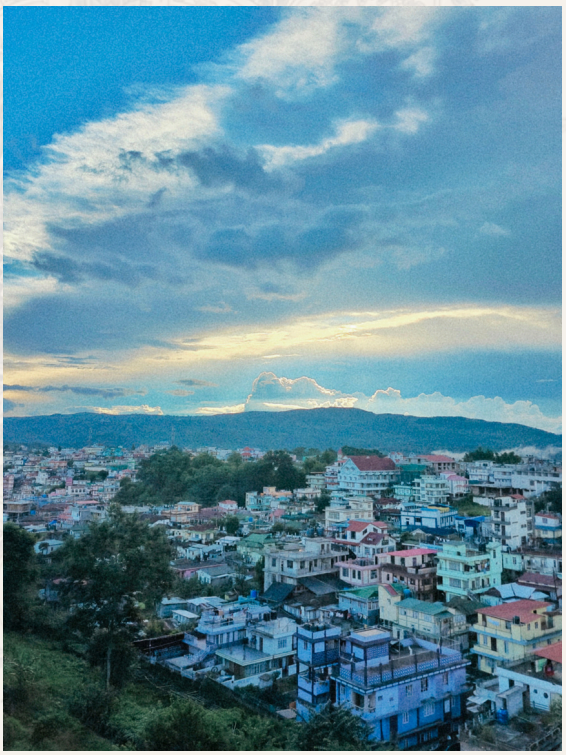
Debjit Kundu, 4th Year, CSE



Debayan Bhattacharya, 4th year, CSE



Sk Asif Hossain, 2nd Year, CSE



Soujanya Mallick , 3rd year, CSE



Garbita Chowdhury, 2nd year, CSE



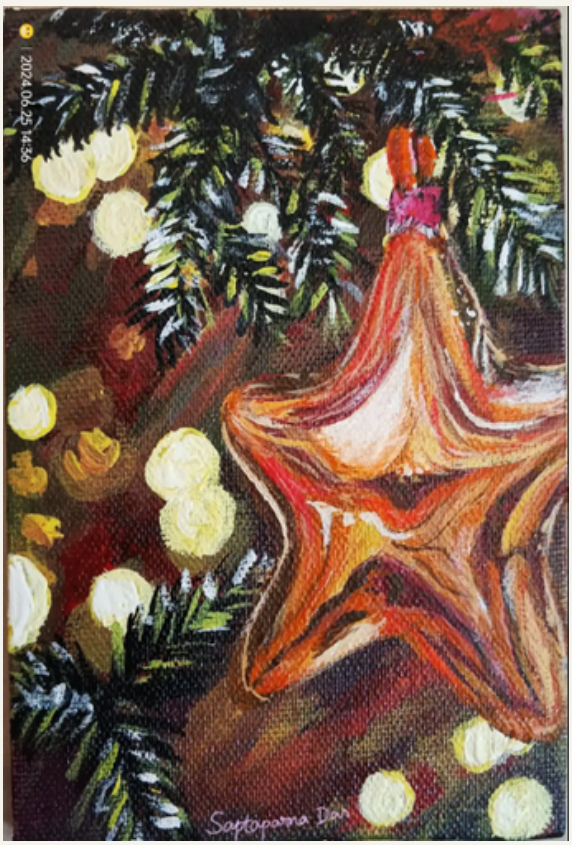
DRAWING



Aman Prasad, 2nd Year, CSE



Anjan Hembrom, 2nd Year, CSE



Saptaparna Das, 3rd Year, CSE



Krishiv Pal, 2nd Year ,CSE



Anirudha Saha, 2nd Year, CSE



Adrija Biswas, 2nd Year, CSE





Utsa Das, 4th Year, CSE



Sreejita Sinha, 2nd Year, CSE



Mahashweta Mondal, 2nd Year, CSE



Nandini Bhagat, 3rd Year, CSE



Samriddha Das, 3rd Year, CSE



UNTIL THE NEXT

ALGORITHM...

</>

PRISMA 3.0

WHERE IDEAS BECOME POSSIBILITIES.

Prisma 3.0 is not just a magazine,
it's a canvas where ideas meet
imagination, where thoughts find voice
and creativity builds a better tomorrow.

KEEP THE SPARK ALIVE