

Programming and Data Structures - I

Lecture 1

Kripabandhu Ghosh

CDS, IISER Kolkata

Introduction to the Instructor



Name: Kripabandhu Ghosh

Highest Qualification: PhD, Indian Statistical Institute, Kolkata (CS)

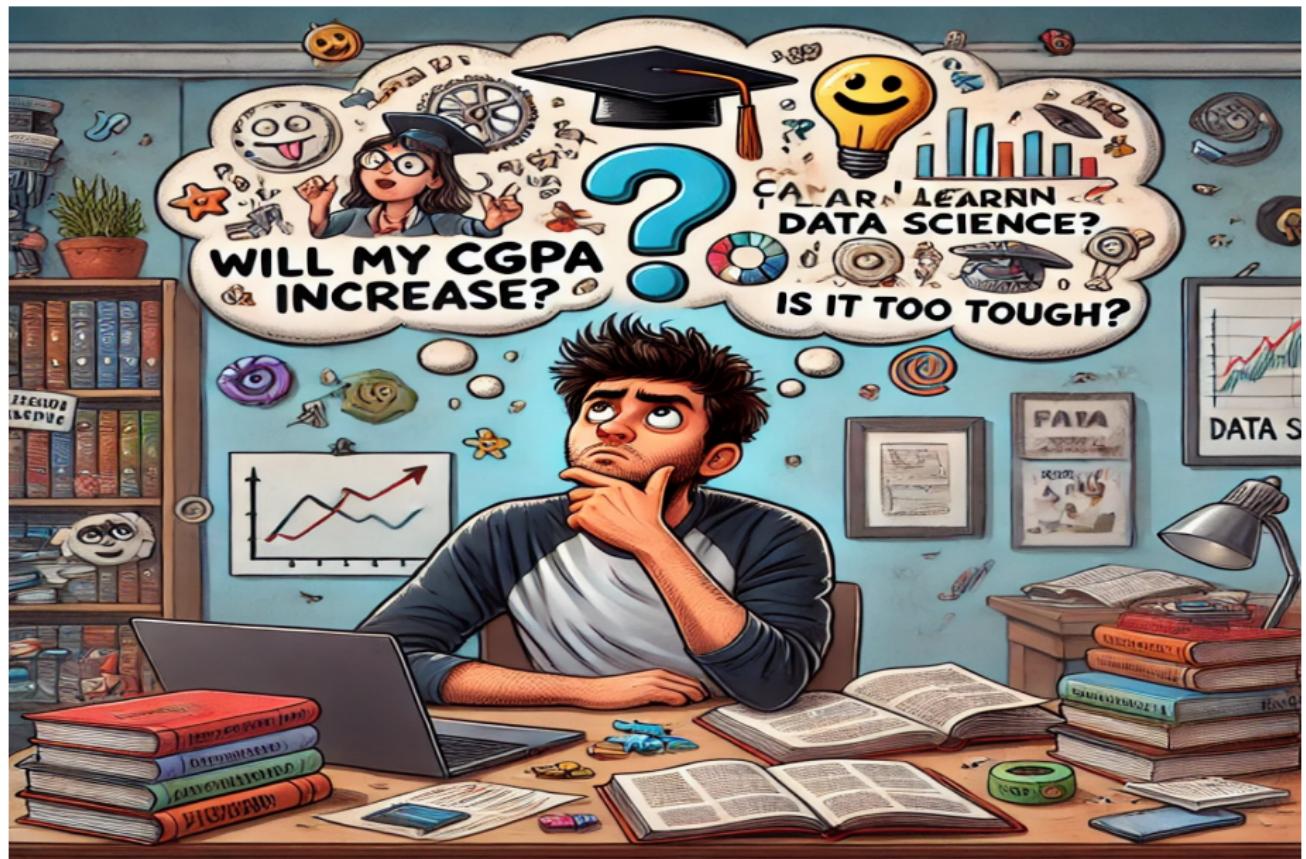
Areas of Interest: Information Retrieval/Data Mining/AI on Legal Domain,
Fairness in Text Summarization, ML, NLP, LLMs

Email: kripaghosh [AT] iiserkol.ac.in

Introduction to the Instructor



About the Course: CS3101



About the course: CS3101

Learning C

- ❖ Starts with a background of C
- ❖ The first C program
- ❖ Basic concepts of C language
- ❖ Plenty of examples to illustrate each concept

Basics of Data Structures and Algorithms

- ❖ Basic data structures: Array, Stack, Queue
- ❖ Basic algorithms: Searching and Sorting
- ❖ Complexity of Algorithms

About the course: CS3101

Learning C

- ❖ Starts with a background of C
- ❖ The first C program
- ❖ Basic concepts of C language
- ❖ Plenty of examples to illustrate each concept

Basics of Data Structures and Algorithms

- ❖ Basic data structures: Array, Stack, Queue
- ❖ Basic algorithms: Searching and Sorting
- ❖ Complexity of Algorithms

All implementations in C

CS3101: Questions to ponder



Our TAs

Coming soon....

Our TAs

Coming soon....



Modus operandi of CS3101

- Theory (Friday: 5–7 PM at LHC 107)
- Lab (Wednesday: 2-5 PM at CC1 and CC2)

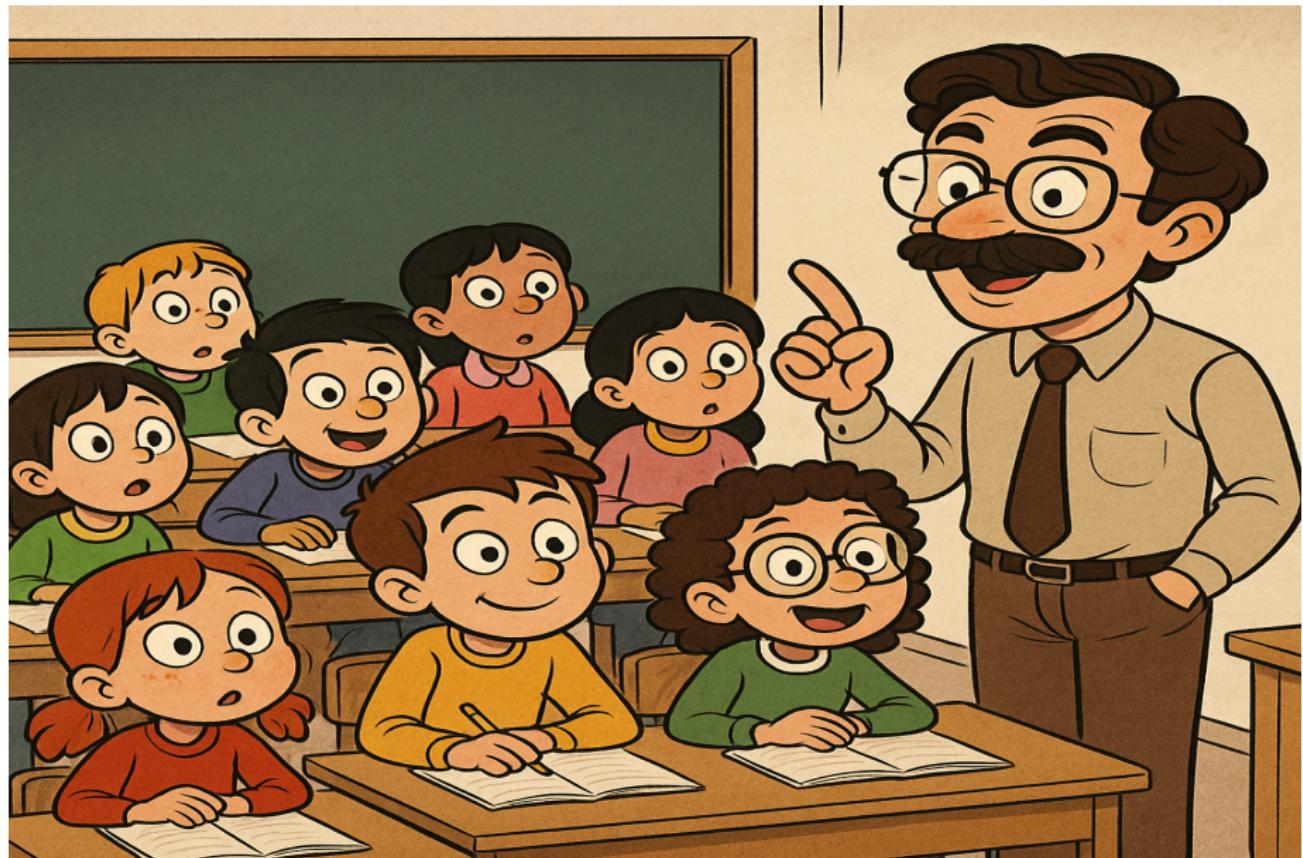
Modus operandi of CS3101

- Theory (Friday: 5–7 PM at LHC 107)
- Lab (Wednesday: 2-5 PM at CC1 and CC2)

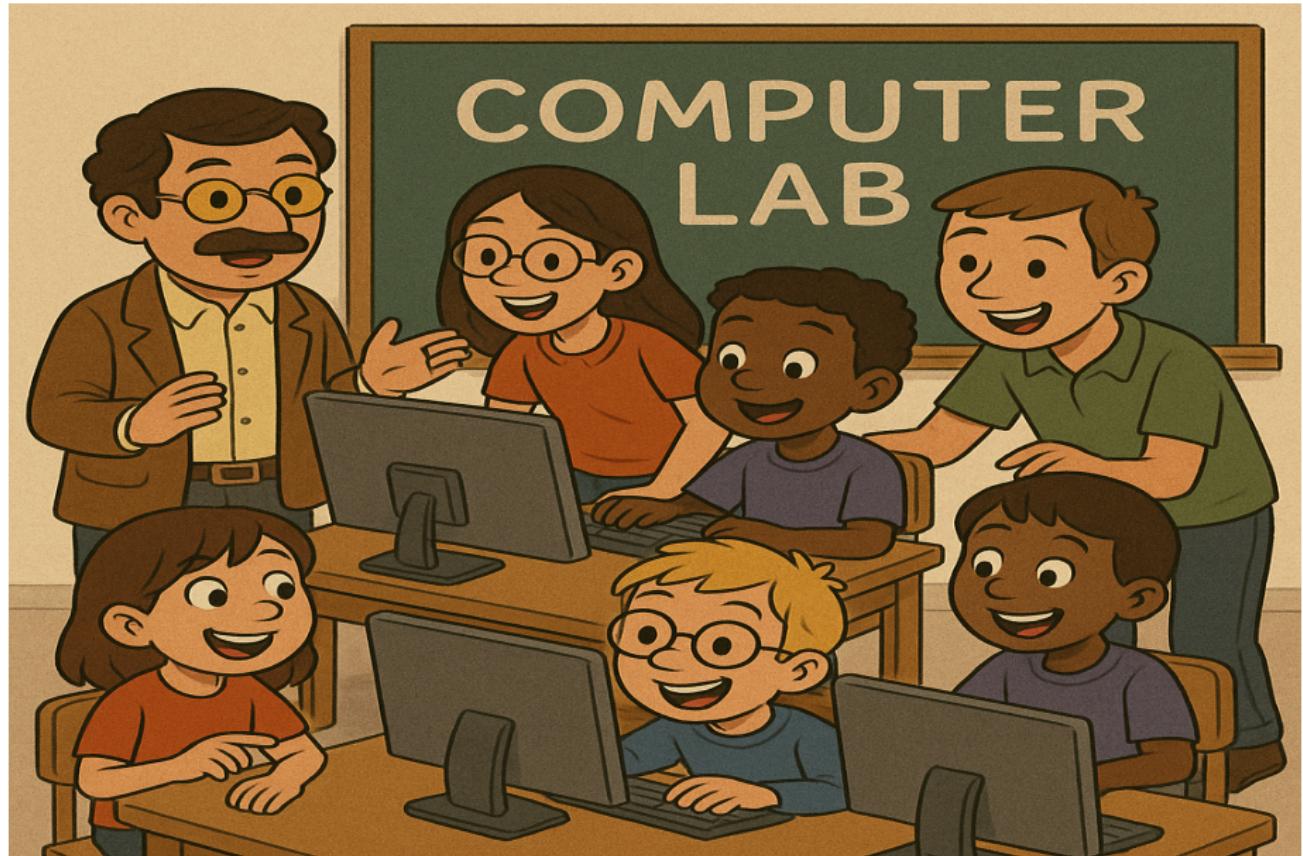
Help us to help you

-  Contact the TAs in case of confusion
-  Ask questions

Theory: Fridays 5–7 PM at LHC 107



Lab: Wednesdays 2-5 PM at CC1 and CC2



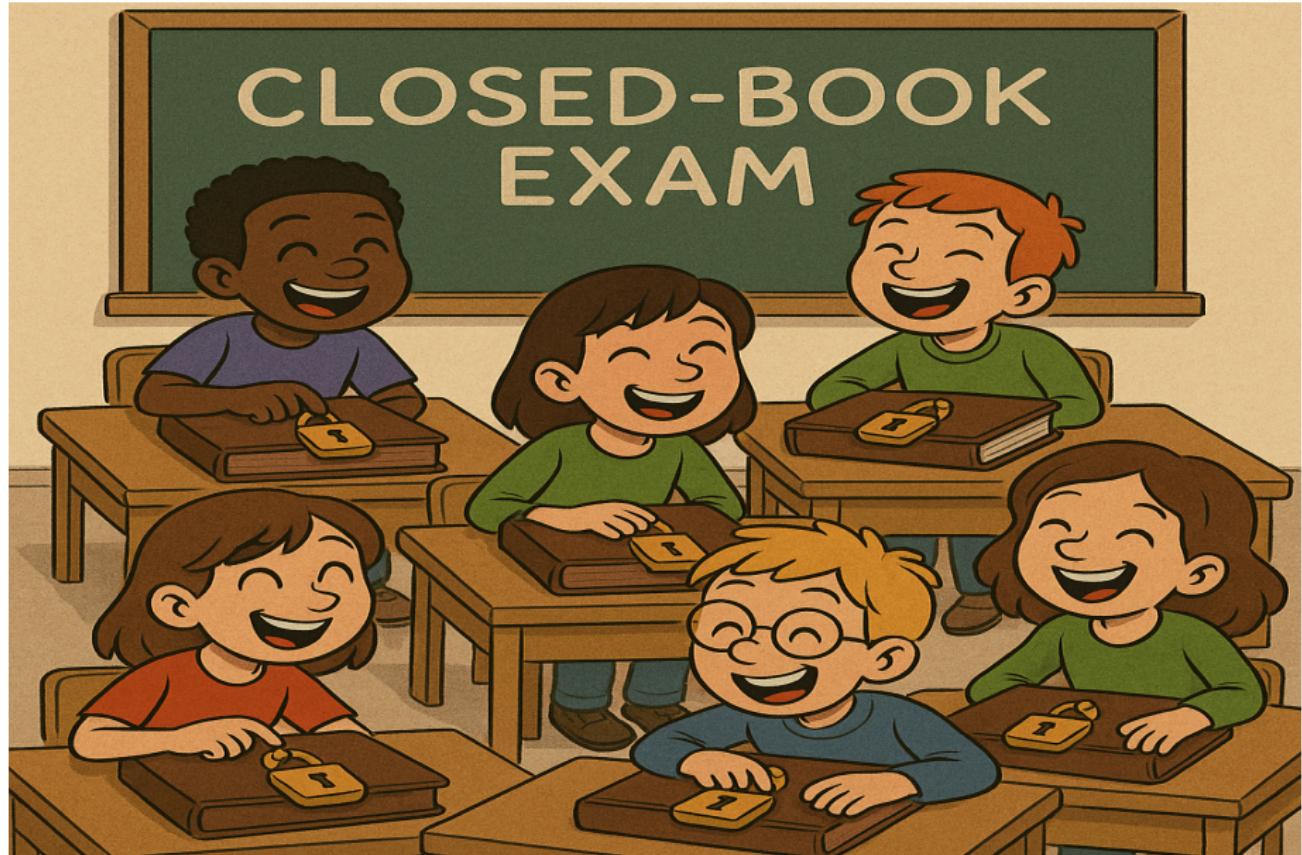
Total marks: 100

- ☛ Quizzes/Class Tests: shorter sets of questions
- ☛ End Sem: longer sets of questions

Evaluation: First 35 marks (approx)



Evaluation: Rest 65 marks (approx)



Some reflections before we start



Some reflections before we start

I do not know what I may appear to the world, but to myself I seem to have been only like a boy playing on the sea-shore, and diverting myself in now and then finding a smoother pebble or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me. – Isaac Newton.

Teaching is the highest form of understanding. – Aristotle

It is the supreme art of the teacher to awaken joy in creative expression and knowledge. – Albert Einstein

I am confident...well almost...

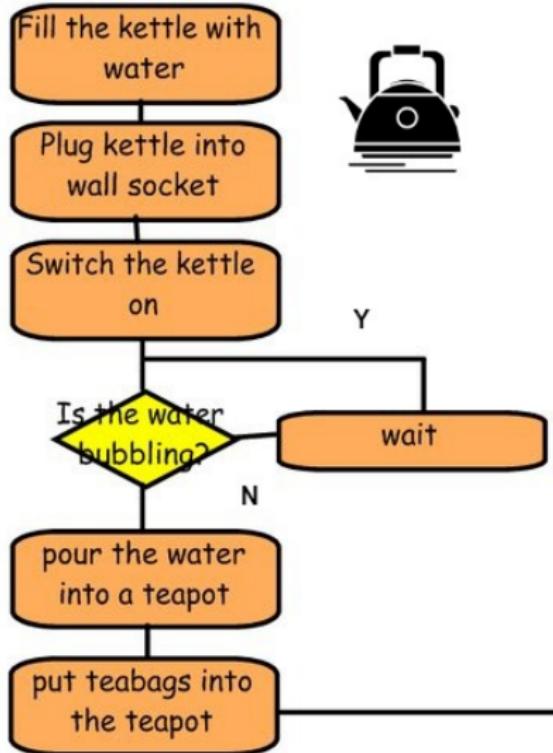


ALGORITHM

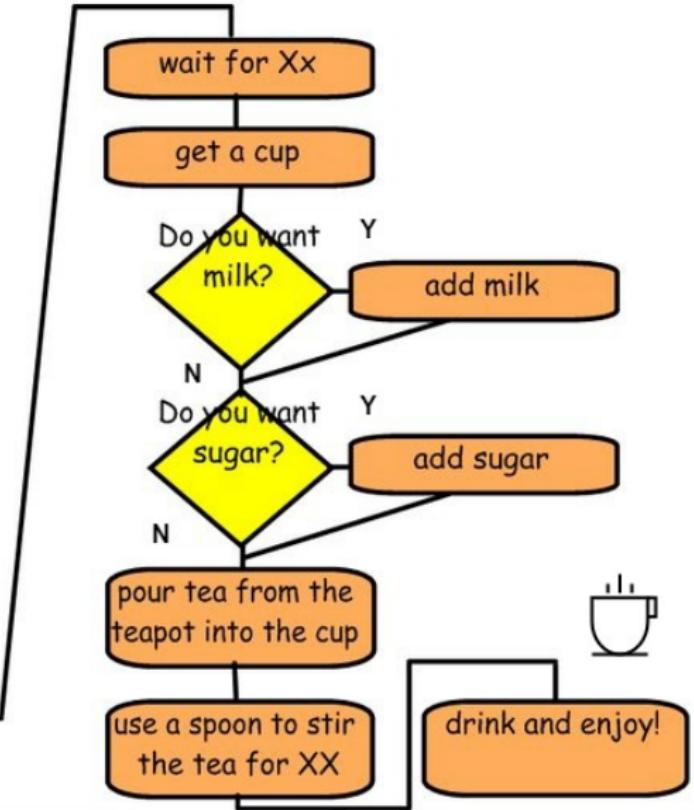
A Cup of Tea



A Cup of Tea¹

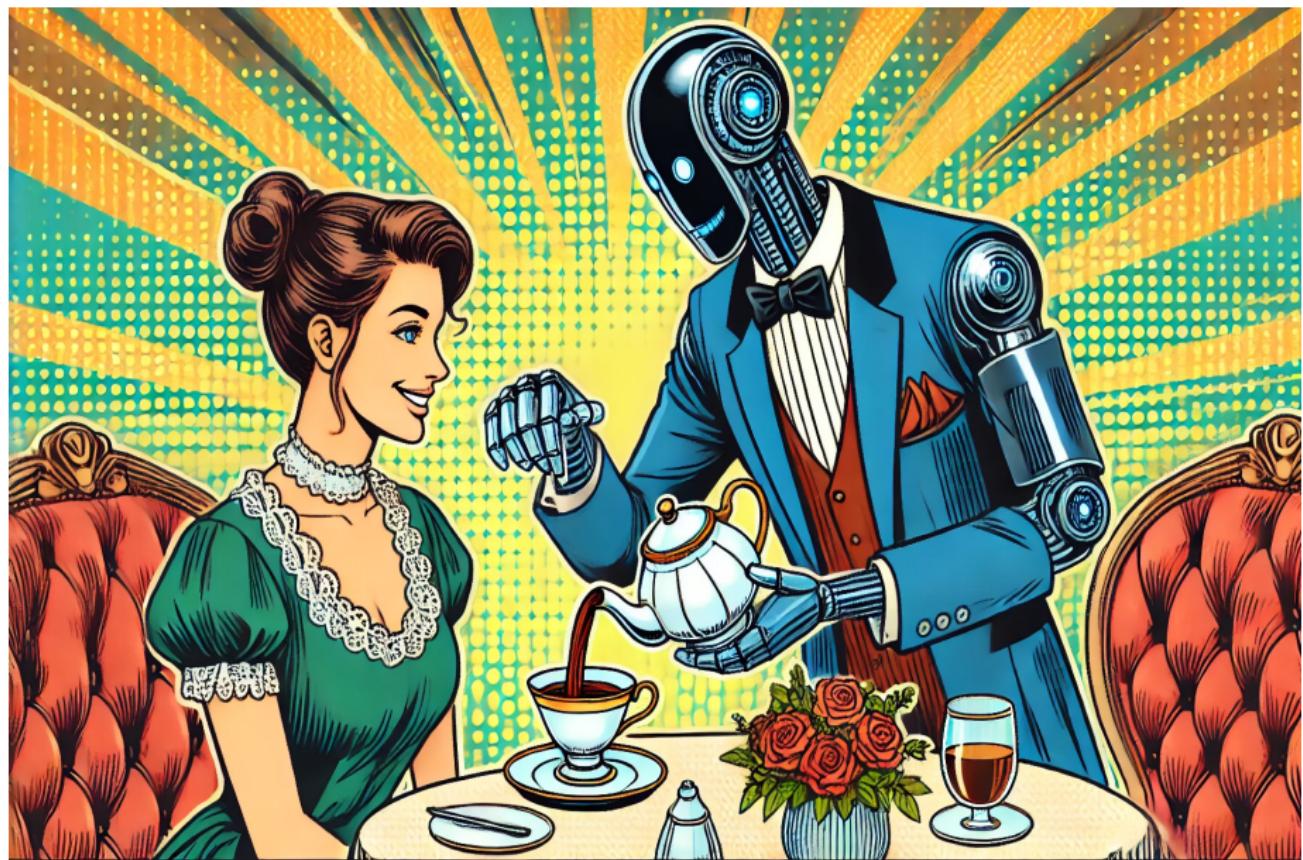


Y



¹<https://x.com/MrLMansfield/status/1086356456734158848>

A Cup of Tea: Automation, Efficiency, Speed, Diligence...



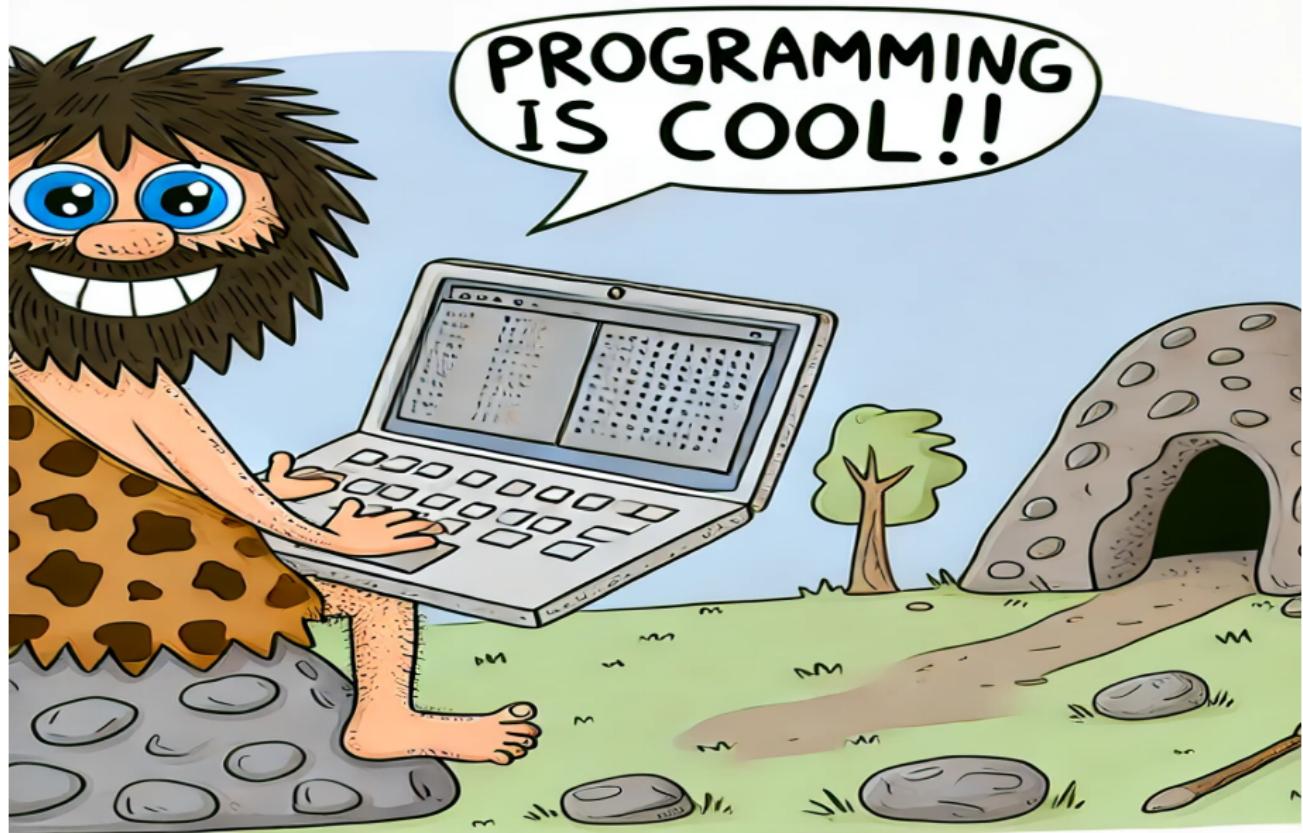
Or more serious problems...



Origin of Programming



Origin of Programming



Agriculture and Farming



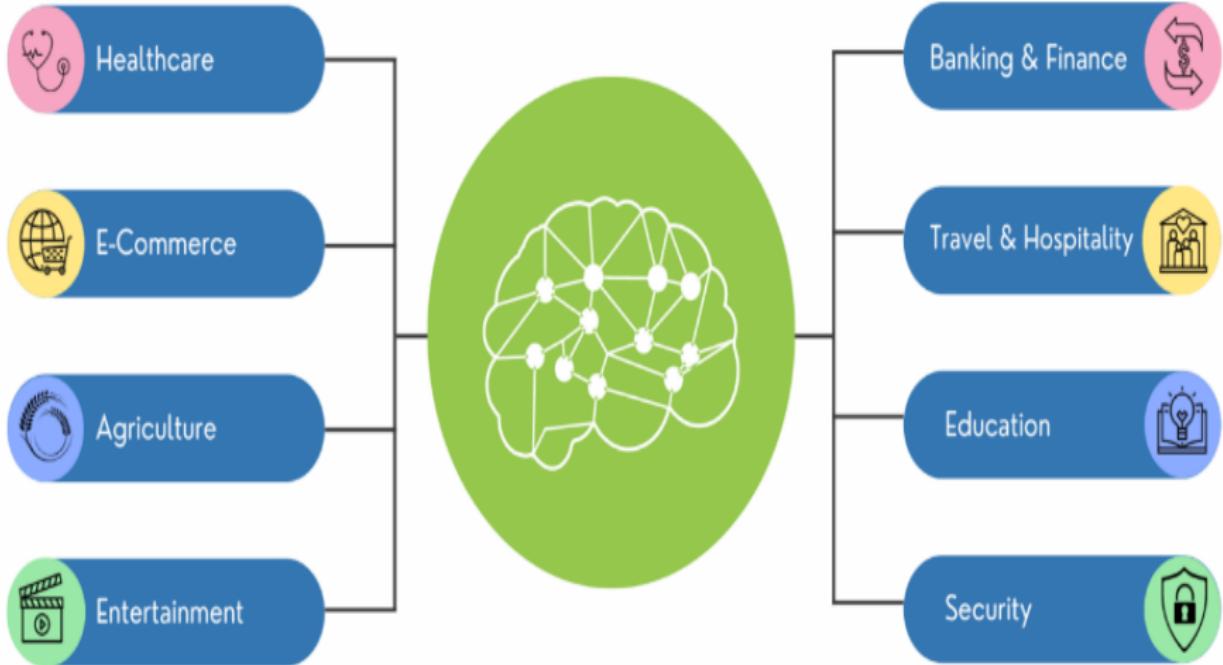
Agriculture and Farming: AI²



²<https://iotechworld.com/>

indian-government-approved-first-agriculture-drone-agribot-uav-drone/

Impactful areas: AI³



³<https://zessta.com/how-ai-holds-an-impact-on-your-business-across-domains/>

algorithm

Origin: *al-Khwārizmi*



Origin⁴



Latinization of the name of the mathematician *Muhammad ibn Musa al-Khwarizmi* (believed to be the inventor of algebra)

⁴Image source: https://en.wikipedia.org/wiki/Muhammad_ibn_Musa_al-Khwarizmi

Origin (contd.)

- Corruption of the term *algorism* (Algorism is the technique of performing basic arithmetic by writing numbers in place value form and applying a set of memorized rules and facts to the digits)^a
- *Algorithmus* (Vollständiges mathematisches Lexicon; Leipzig, 1747) refers to four types of arithmetic calculations, viz. addition, multiplication, subtraction and division
- Latin phrase *algorithmus infinitesimalis* means “ways of calculation with infinitely small quantities”

^a<https://tinyurl.com/algorismwiki>

The term Algorithm gained popularity during 1950 after being used in Euclid's algorithm for finding the greatest common divisor of two numbers (Euclid's *Elements*; Book 7, Propositions 1 and 2)

Definition⁵

Algorithm

Any *well-defined* computation procedure that takes some value, or set of values, as *input* and produces some value, or set of values, as *output*

An algorithm is a sequence of computational steps that transform the input to the output

⁵Introduction to Algorithms. Cormen et al. Chapter 1, 3rd edition.

Example Algorithm

Algorithm

To find the *maximum* of n numbers stored in A

Input

n numbers stored in A

Output

The maximum of n numbers stored in A

Example Algorithm (contd.)

Steps

- S1: Let $i = 0$
- S2: Let $Max = A[0]$
- S3: While($1 \leq i \leq n - 1$)
 - S4: If $A[i] > Max$, $Max = A[i]$
- S5: Output Max

Features of an Algorithm⁶

- ① Finiteness: An algorithm must always terminate after a finite number of steps.
- ② Definiteness: Each step of an algorithm must be precisely (rigorously and unambiguously) defined.
- ③ Input: An algorithm has zero or more inputs
- ④ Output: An algorithm has one or more outputs.
- ⑤ Effectiveness: All the operations must be done using pencil and paper.

⁶The Art of Computer Programming. Donald E. Knuth. Vol 1, 3rd Ed.

Donald E. Knuth⁷



- Knuth has been called the “father of the analysis of algorithms”
- Author of TEX, LR parser etc.
- Turing award winner

⁷Image source https://en.wikipedia.org/wiki/Donald_Knuth

Features of an Algorithm: Finiteness

Desideratum

- An algorithm must always terminate after a finite number of steps
 - The example algorithm will terminate in exactly n steps
 - The While loop will terminate in n iterations

Exceptions

- Find the maximum in the set of all natural numbers (1, 2, ...)
- Find all the irrational numbers in the interval (0, 1)
- An infinite loop (e.g. While(1))
 - It is possible to write an infinite loop in a programming language (e.g. C)
 - It is prohibited to write such a loop intentionally as it will not lead to a meaningful algorithm

Features of an Algorithm: Definiteness

Desideratum

- Each step of an algorithm must be precisely (rigorously and unambiguously) defined
 - All the steps in the example algorithm satisfy definiteness.
 - Computer languages or programming languages are defined to ensure definiteness.

Exceptions

- Ask a robot to prepare tea with imprecise instructions like *add a bit of sugar*, *boil the water enough* etc.

Features of an Algorithm: Input

Desideratum

- An algorithm has zero or more inputs.
 - In the example algorithm there are n inputs.
 - There may not be any input (e.g. print your name)

Features of an Algorithm: Output

Desideratum

- An algorithm has one or more outputs.
 - In the example algorithm there is *one* output.
 - In our case, the output is also correct.

Features of an Algorithm: Effectiveness

Desideratum

- All the operations must be sufficiently basic so that they can be done exactly and in a finite number of steps using pencil and paper.
 - The example algorithm satisfies the effectiveness criterion.
 - The maximum value can be calculated using this algorithm using pencil and paper.

Exceptions

- Find the maximum irrational number.
- Perform Step 4 if “4 is the largest integer n for which there is a solution to the equation $w^n + x^n + y^n = z^n$ in positive integers w, x, y and z ”.

Example Algorithm (repeated)

Steps

- S1: Let $i = 0$
- S2: Let $Max = A[0]$
- S3: While($1 \leq i \leq n - 1$)
 - S4: If $A[i] > Max$, $Max = A[i]$
- S5: Output Max

Practical Features of an Algorithm

Desideratum

- Length of time taken to execute the algorithm – number of times each step is executed (time complexity)
- Space (memory) taken by the algorithm (space complexity)

Analysis of the example algorithm

- Time complexity: $O(n)$ (loosely speaking, number of comparisons in the while loop)
- Space complexity: $\approx n$ (size of the set A)

DATA STRUCTURES

Definition

Data structure

A *data structure*^a is a way to store and organize data in order to facilitate access and modifications

^aIntroduction to Algorithms. Cormen et al. Chapter 1, 3rd edition.

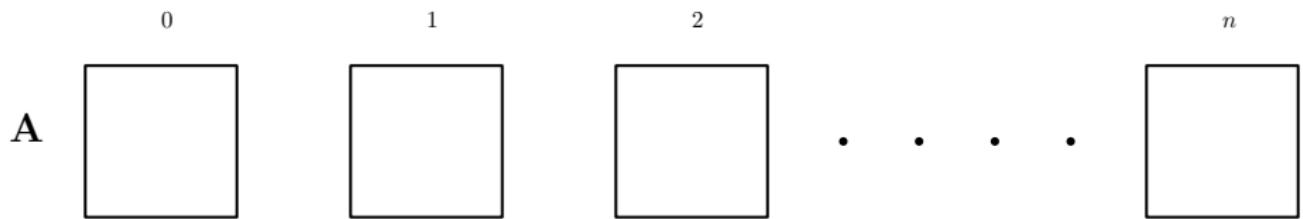
Also called *Information Structures*^a – information with structural relationships between data elements.

^aThe Art of Computer Programming. Donald E. Knuth. Vol 1, 3rd Ed.

Constituents

- Data values (with domains e.g. natural numbers)
- Operations defined on the data

Array



Array

Data structure

In the example, A is a static array of n elements (can also be dynamic)

Constituents

- Data values : integer (range depending upon the programming language)
- Operations defined on the data:
 - Access: Search, Max, Min etc.
 - Modification: Insertion, Deletion

Stack



Queue



Queue: other types



PROGRAMMING LANGUAGES

Programming Languages

- An algorithm can be written in pseudo code or even in (human) natural language
 - Computer language or Programming Language allows the algorithm to be written in machine-readable format
-
- A good programming language ensures Definiteness criterion (also Input and Output)
 - Finiteness and Effectiveness should be ideally ensured by the human programmer

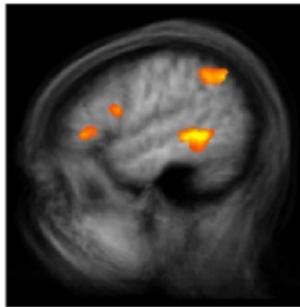
Use of Programming Languages

“Coding is not just code, that is a live thing to serve everyone!” – Ming Song

To enslave the tireless, fast and accurate monster before it becomes your master!

Cognitive Impact of Programming Languages⁸

In 2014, researchers at the University of Passau, University of Magdeburg, Carnegie Mellon, Georgia Tech, the Leibniz Institute for Neurobiology, and the Metop Research Institute investigated how exactly computer programmers understand code using fMRI scans of the 17 programmers' brains



The study found activations of parts of the brain associated with language processing, memory, attention, and logic.

Some other studies argue that programming activates regions responsible for complex cognitive tasks, such as solving math problems, more than language processing.

⁸Understanding Understanding Source Code with Functional Magnetic Resonance Imaging.
Siegmund et al. ICSE 2014. (Core A*)

References

- Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language*, Prentice Hall of India.
- E. Balaguruswamy, *Programming in ANSI C*, Tata McGraw-Hill.
- Yashavant P. Kanetkar, *Let Us C*, 5th Ed.
- Donald E. Knuth *The Art of Computer Programming*, Vols 1 – 3.
- Cormen et al, *Introduction to Algorithms*, 3rd edition.
- Langsam, Augenstein, Tenenbaum, *Data Structures Using C and C++*.
- S.K.Srivastava, Deepali Srivastava, *C In Depth*.



THANKS

YOU