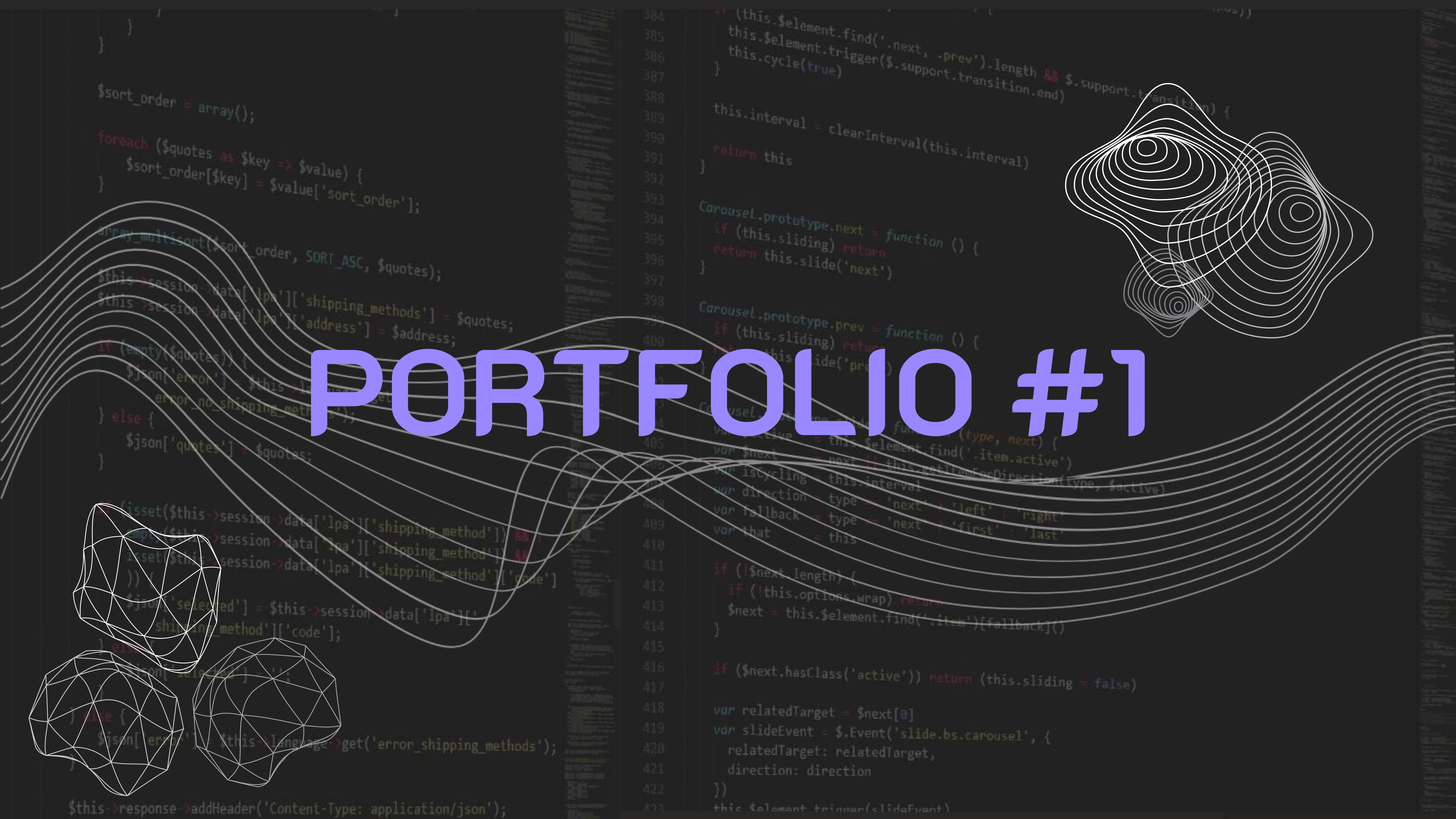
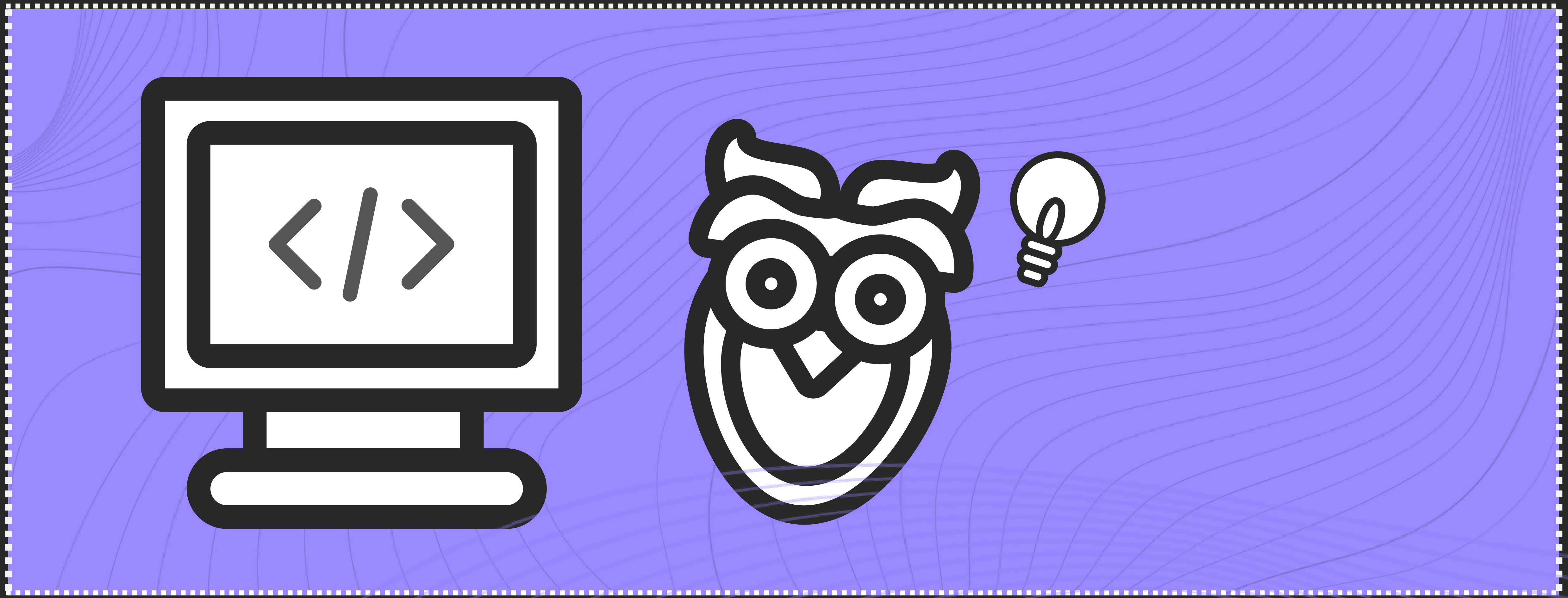
PORTFOLIO #1

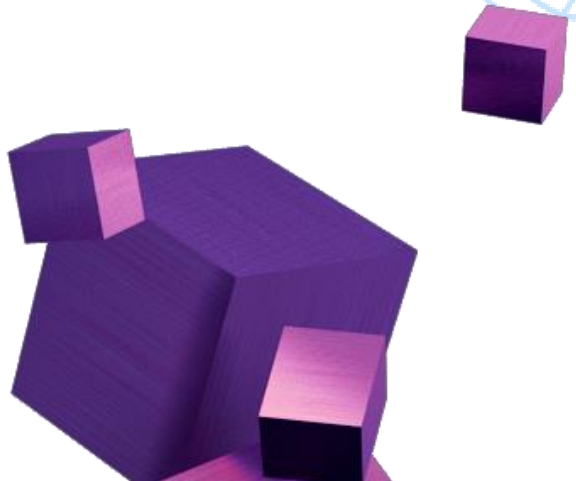


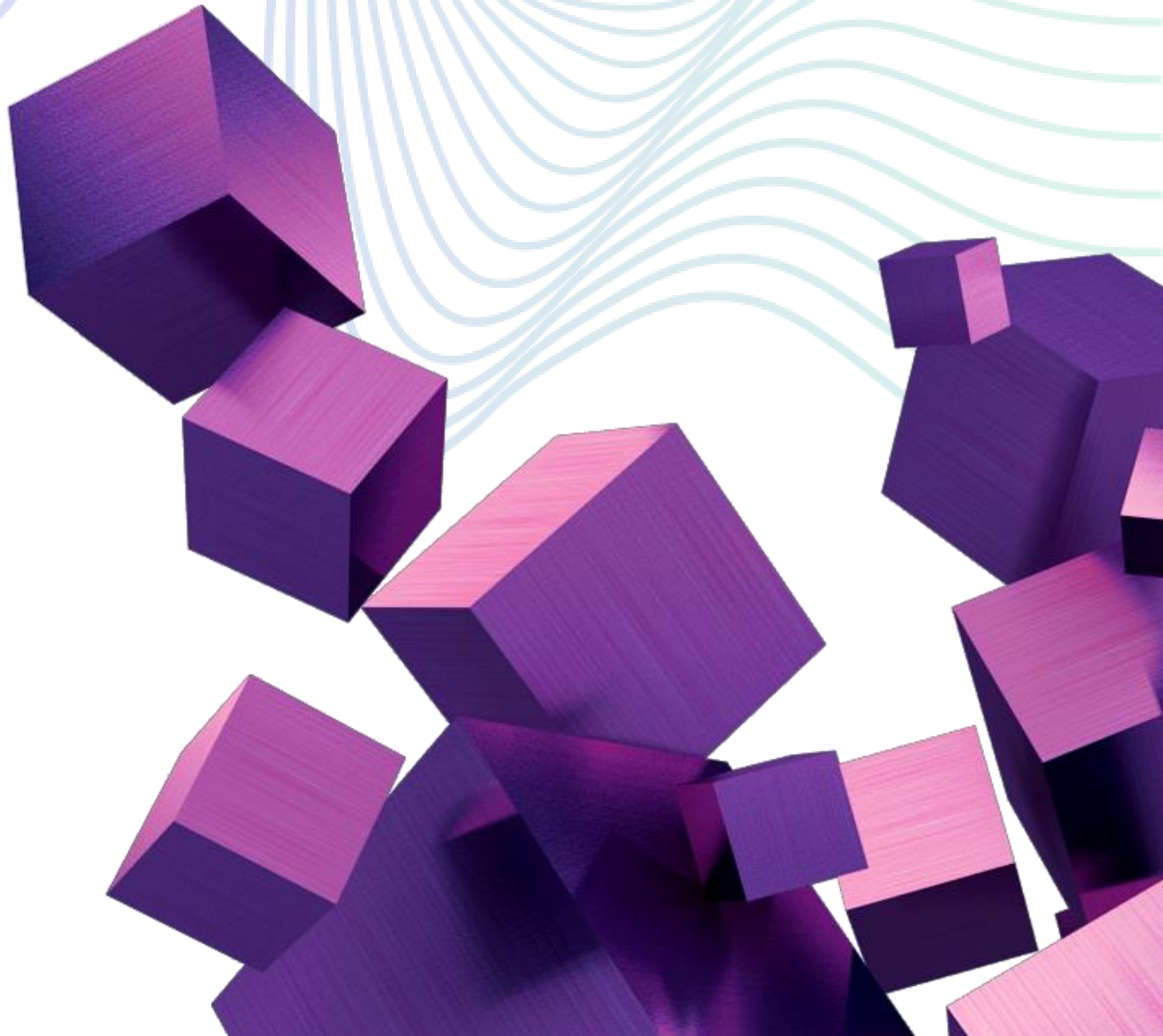
Computer Science as Discipline



Introduction

This provides a comprehensive overview of key sources that discuss various aspects of computer science as a discipline. These sources cover theoretical foundations, educational frameworks, and debates about the nature and scope of computer science.

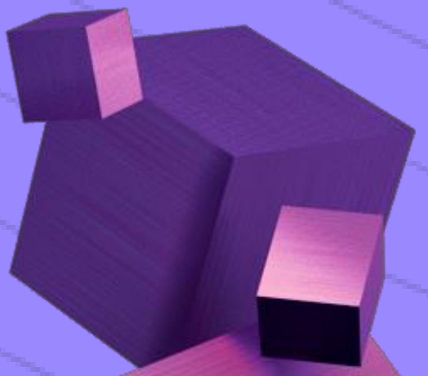


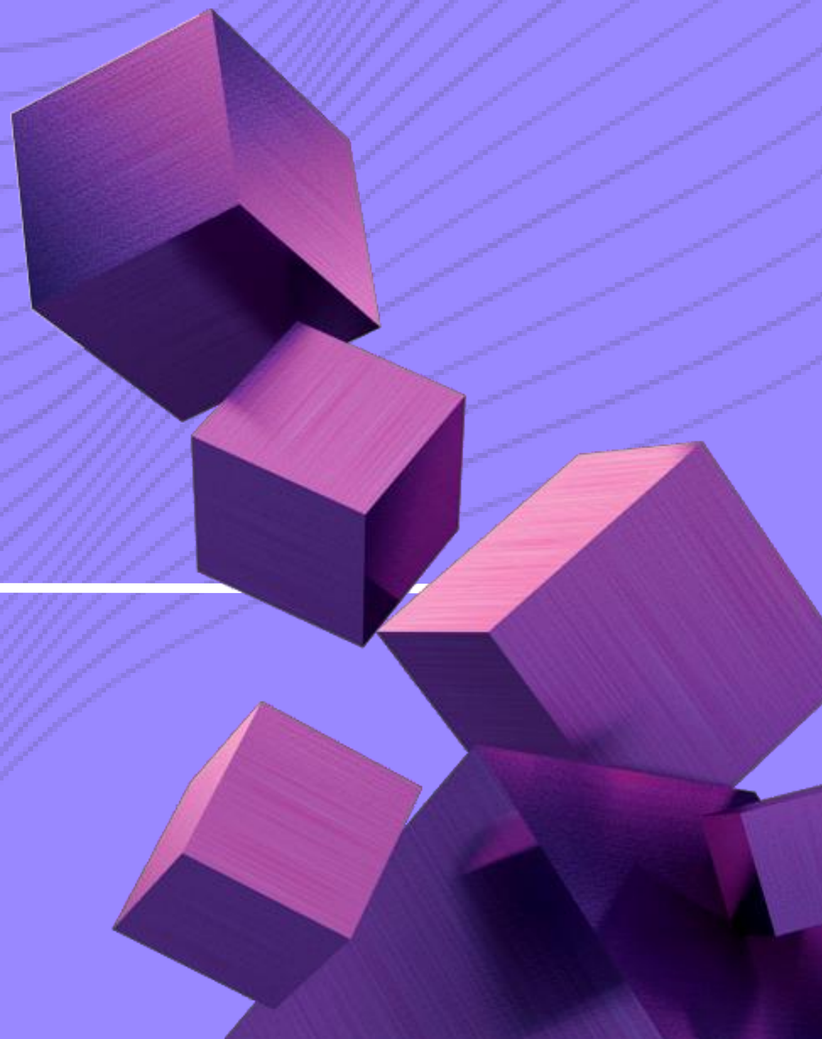


Definition and Scope of Computer Science

Overview

- The examination of what constitutes computer science, including its core principles and areas of study.
- Historical Context is the evolution of the field from its origins to modern developments, highlighting major milestones and influential figures.





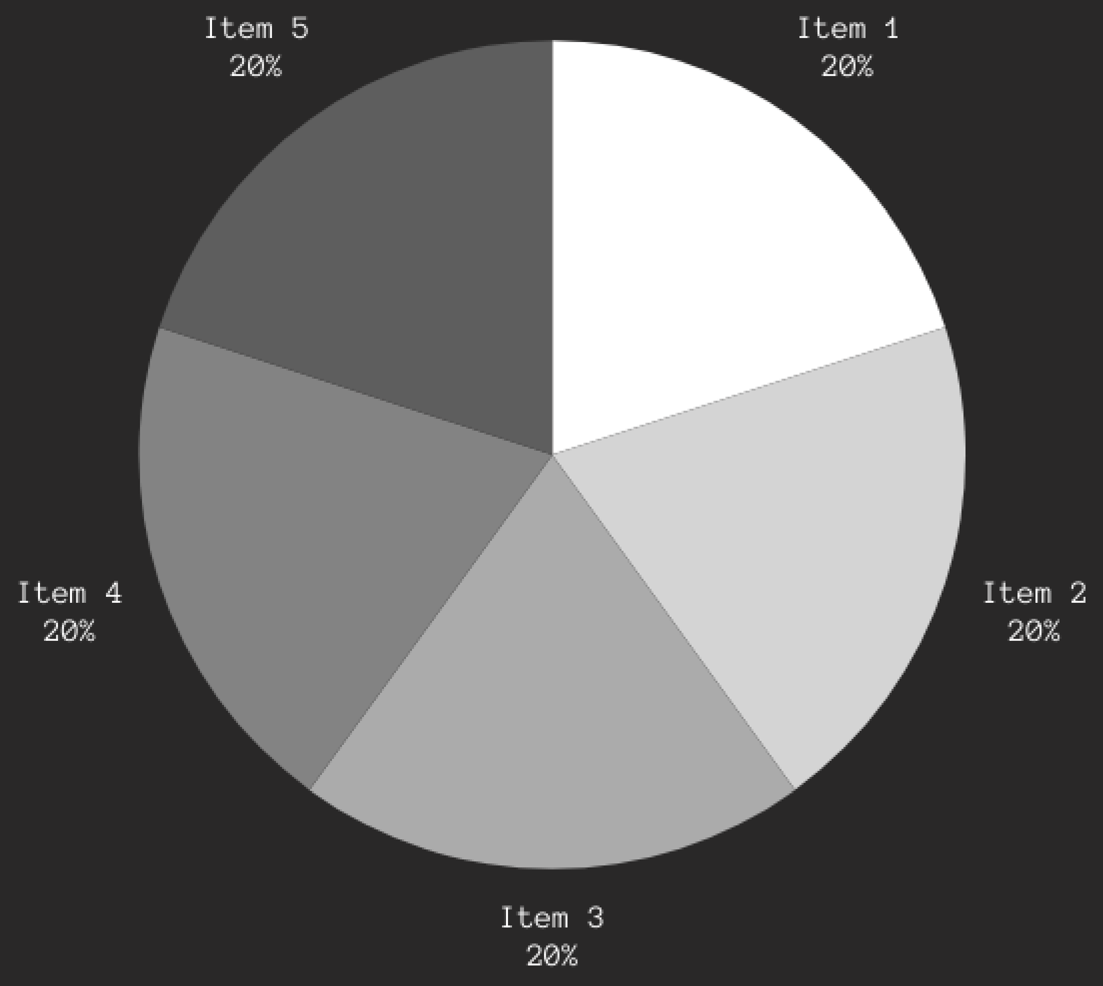
Theoretical Foundations

Algorithms and Complexity

- Algorithm Design are strategies for designing efficient algorithms.
- Computational Complexity, the understanding of time and space complexity and the classification of problems.

Formal Languages and Automata Theory

- Formal Languages is the study of syntax and semantics.
- Automata Theory is the exploration of computational models like finite automata and Turing machines.



Computer science (CS) is the study of algorithms, data structures, and the principles underlying the design and analysis of computer systems. It encompasses a broad range of topics, including software development, hardware design, artificial intelligence, and human-computer interaction.

At its core, computer science seeks to understand and create computational processes and systems that solve problems, optimize performance, and enable innovations across various domains. It integrates theoretical concepts with practical applications, driving advancements in technology and influencing numerous aspects of modern life.

The discipline of computing is the systematic study of algorithmic processes that describe and transform information: their theory, analysis, design, efficiency, implementation, and application.

• PARADIGMS FOR THE DISCIPLINE

The **three major paradigms**, or cultural styles, by which we approach our work provide a context for our definition of the discipline of computing.

- The **rationalist paradigm**, which was common among theoretical computer scientists, defines computer science as a branch of mathematics, treats programs on a par with mathematical objects, and seeks certain, a priori knowledge about their 'correctness' by means of deductive reasoning.
- The **technocratic paradigm**, promulgated mainly by software engineers, defines computer science as an engineering discipline, treats programs as mere data, and seeks probable, a posteriori knowledge about their reliability empirically using testing suites.
- The **scientific paradigm**, prevalent in the branches of artificial intelligence, defines computer science as a natural (empirical) science, takes programs to be entities on a par with mental processes, and seeks a priori and a posteriori knowledge about them by combining formal deduction and scientific experimentation.

- **The** most consistent aspect of computer science is change. It is not surprising then, to note a significant change in the ability and interest of computer science departments to offer new courses for liberal arts and sciences students.

- **Knowing** the philosophical grounding of the discipline, such as the nature of computation, algorithmic theory, and ethics brought about by technology, is important. The inclusion of philosophical discussions in the curriculum develops a broader insight into one's discipline, enriching a person's critical thinking and ethical awareness.

- **Computer** science can contribute in significant ways to all of the experiences. Consequently it can play a vital role as a discipline in the core of the liberal arts and sciences.

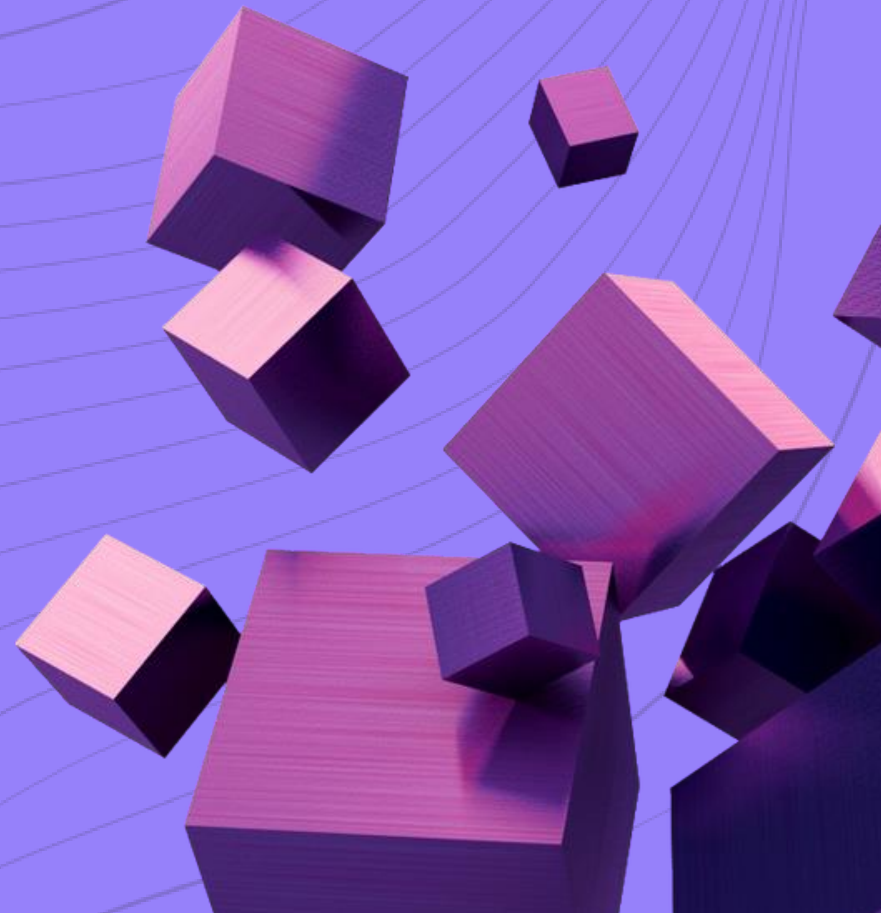
Historical Context

The historical setting in computer science would then include early theoretical conceptions from the 19th century by Charles Babbage's mechanical computer and Alan Turing's formalization of computation. In the middle of the 20th century, the discipline took a practical turn with the development of electronic computers and programming languages. Further work allowed for the emergence of personal computing, microprocessors, and the internet, finally securing a place among the sciences for computer science, which is still developing today, thanks to contributions from artificial intelligence and data science.

Summary

Computer science is the study of algorithms, data structures, and principles in computer systems, encompassing software development, hardware design, artificial intelligence, and human-computer interaction. It aims to understand and create computational processes that solve problems, optimize performance, and enable innovations. It integrates theoretical concepts with practical applications, driving technological advancements and influencing modern life.

Understanding the philosophical grounding of computer science is crucial for developing critical thinking and ethical awareness. Computer science can contribute significantly to liberal arts and sciences.



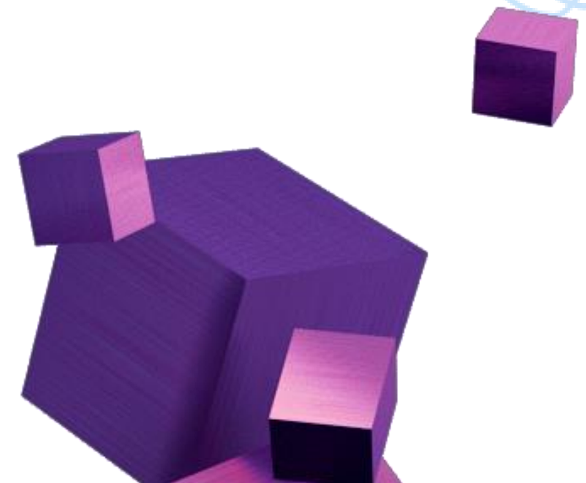
Key Topics

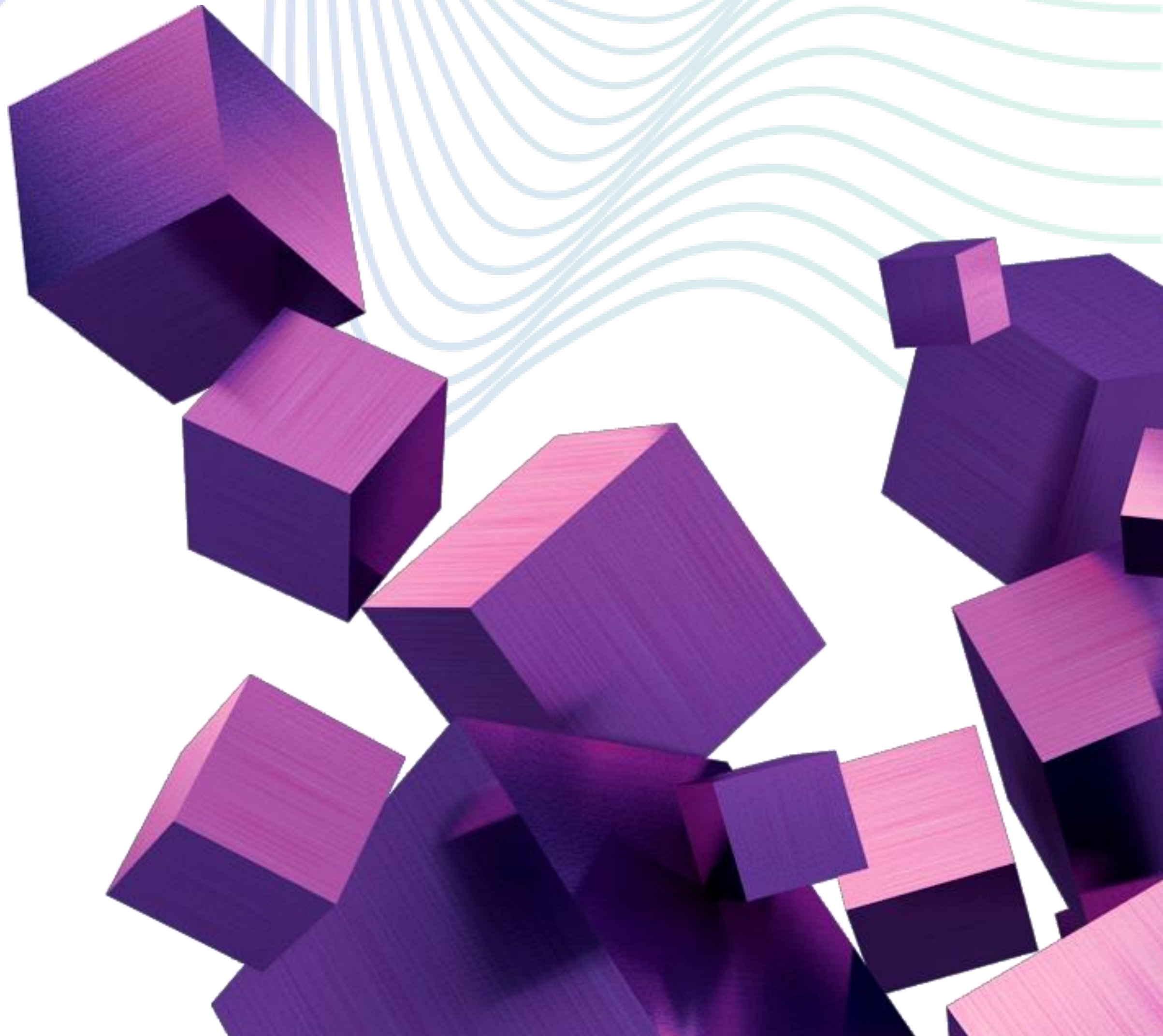
- Core Discipline Argument
- Interdisciplinary Integration
- Computer Science
- Theoritical Foundation
- The Three Paradigms
- Historical Context

5 Computing Disciplines and Majors

Introduction

This provides a comprehensive overview of that discuss various aspects of the five computing and major disciplines.



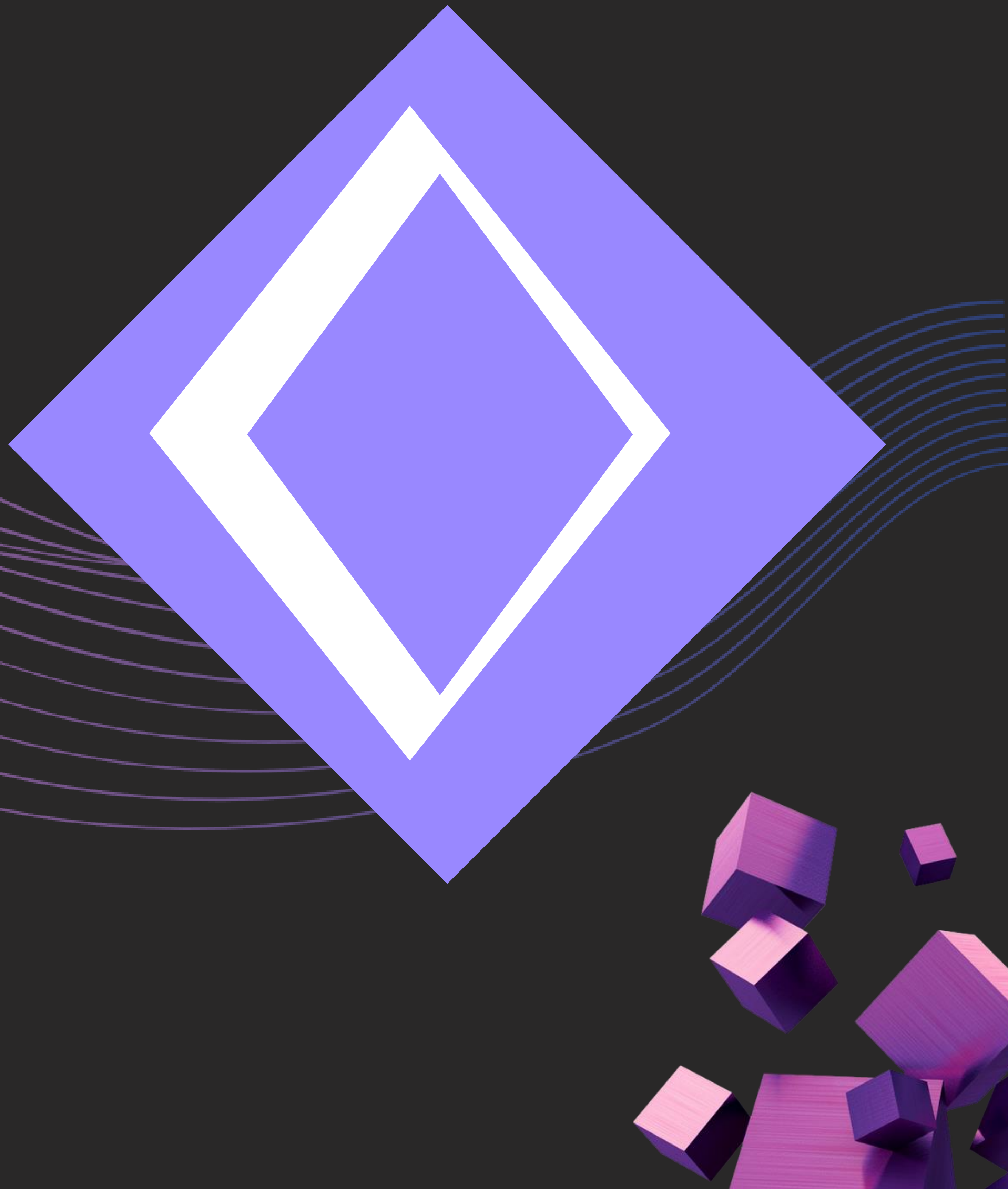


The Five Major Computing fields of Computer Science (CS), Information Systems (IS), Information Technology (IT), Software Engineering (SE) and Computer Engineering (CE) differ in their approach towards computing.

- **Computer Science (CS)** research includes studying computation and its possibilities, designing and implementing algorithms, and various programming mechanisms (Smith, 2024).

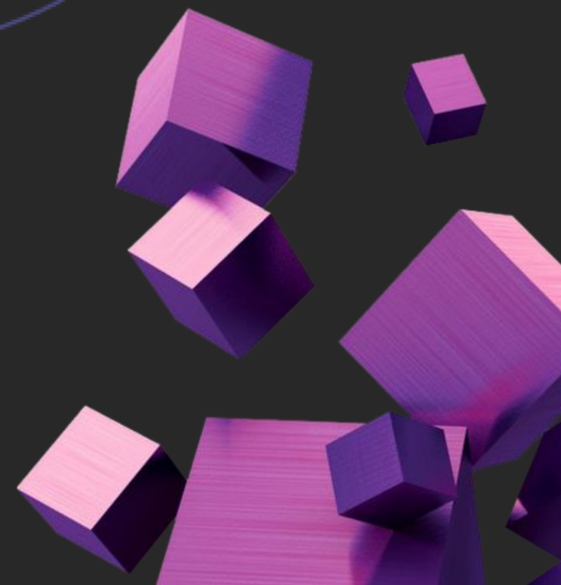
- **Information Technology (IT)** includes the use of technology, controlling and running of systems and networks and providing backup systems including safety and security. (Roy et al., n.d.)

- **Information Systems (IS)** are critical components that assist organizations in achieving their strategic and operational goals. (Tay, M. H., n.d.)



- **Software Engineering** (SE) concerns itself with the principled approach to the development, operation, and maintenance of software and systems, including design, programming, testing, and project management (Mall, 2020).

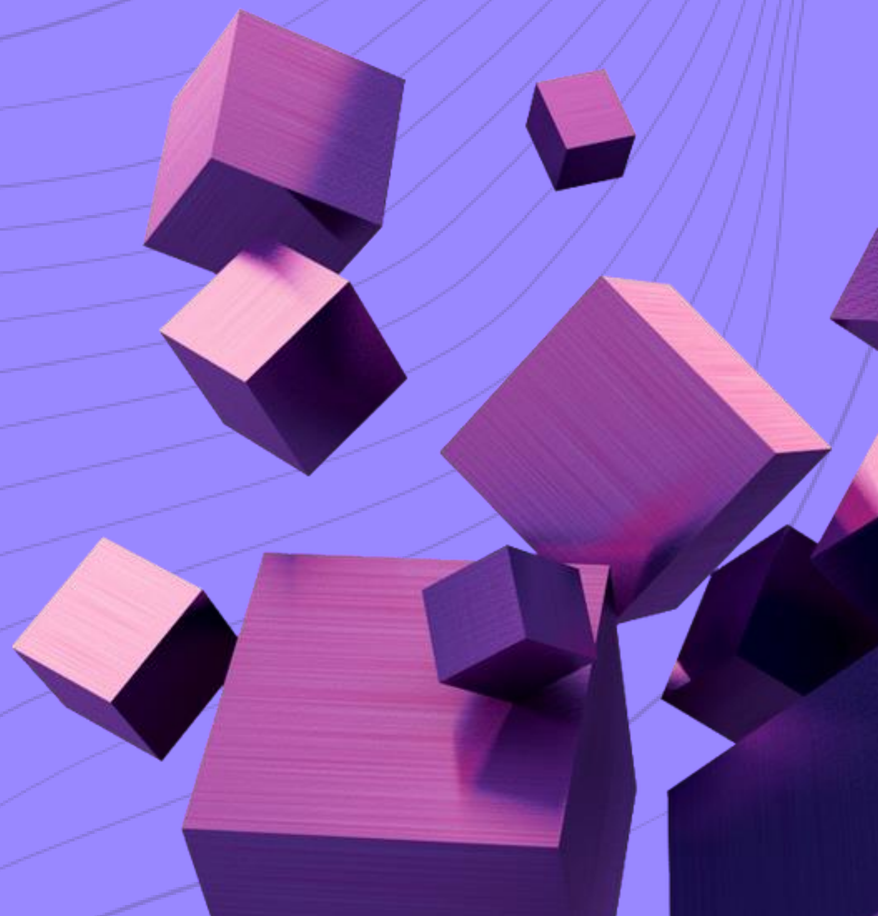
- **Computer Engineering** (CE) is an integration of microelectronics and computing to build and improve computer components and systems which cover the architecture and software of computers.



Summary

Computer Science (CS) explores theoretical and practical aspects of computation and algorithms; **Information Technology** (IT) focuses on managing and implementing technology systems and networks; **Information Systems** (IS) integrates technology with organizational processes to achieve business goals; **Software Engineering** (SE) employs systematic approaches to develop, maintain, and test software; and **Computer Engineering** (CE) combines microelectronics and computing principles to design and improve hardware and software systems.

All these fields are essential in developing technology as they all benefit one another and help with the process in creating and managing anything digital.



References

- Denning, P., Comer, D., Gries, D., Mulder, M., Tucker, A., Turner, J., & Young, P. (1989). Computing as a Discipline. Computer, 22, 63-70. DOI: 10.1109/2.19833

- Beck, R. E., Cassel, L. N., & Austing, R. H. (1989b). Computer Science: A Core Discipline of Liberal Arts and Sciences. Computer Science: A Core Discipline of Liberal Arts and Sciences. DOI: 10.1145/65293.65300

- Jubaer, S. (2021). An introduction to computer science and its history. ResearchGate. Clark, M. (2003). Computer Science: A Hard-Applied Discipline? Teaching in Higher Education, 8(1), 71–87. DOI: 10.1080/1356251032000052339

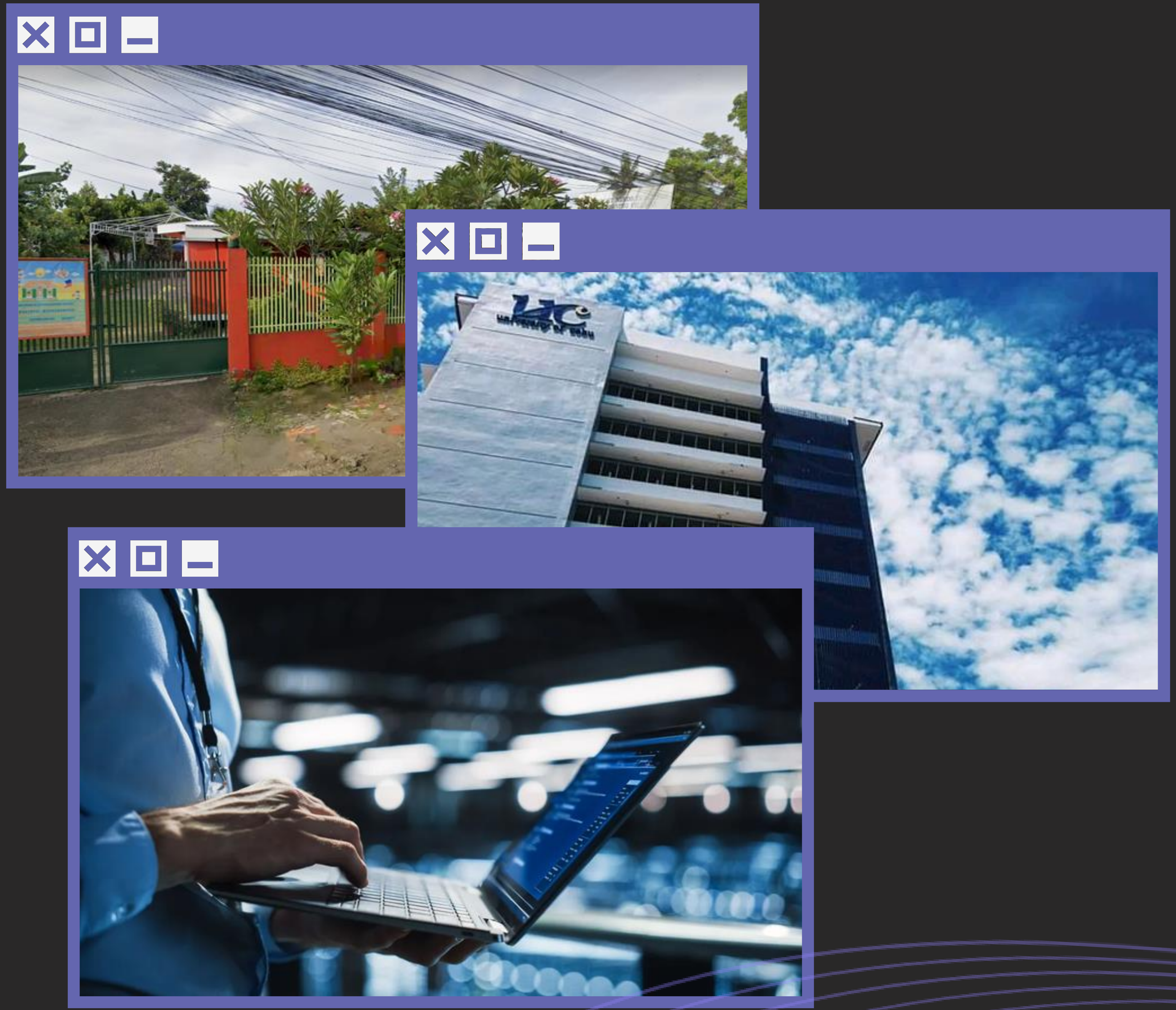
- Tedre, M. (2007). Know Your Discipline: Teaching the Philosophy of Computer Science. Journal of Information Technology Education: Research, 6(1), 105-122. Informing Science Institute

- Eden, A. (2007). The three paradigms of computer science. Retrieved from https://eclass.hmu.gr/modules/document/file.php/TP374/Lecture%20Presentations/L1%20-%20 Introduction/Literature/2007%20-%20The%20 three%20paradigms%20of%20computer%20science%20-%20A.%20Eden.pdf

References

- Computing: An introduction to computer science and programming. (n.d.). Retrieved August 30, 2024, from Denning, P., Comer, D., Gries, D., Mulder, M., Tucker, A., Turner, J., & Young, P. (1989). Computing as a Discipline. Computer, 22, 63-70. DOI: 10.1109/2.19833
- Tay, M. H. (n.d.). Fundamentals of information systems (5th ed.). Radford University. Beck, R. E., Cassel, L. N., & Austing, R. H. (1989b). Computer Science: A Core Discipline of Liberal Arts and Sciences. Computer Science: A Core Discipline of Liberal Arts and Sciences. DOI: 10.1145/65293.65300
- Roy, S., Daniel, C., & Agrawal, M. (n.d.). Fundamentals of information technology. Digital Commons @ University of South Florida. https://digitalcommons.usf.edu/dit_tb_eng/19/
- Department of Computer Science and Engineering. (2019). Undergraduate handbook: Computer engineering (Version 2). Penn State University. Tedre, M. (2007). Know Your Discipline: Teaching the Philosophy of Computer Science. Journal of Information Technology Education: Research, 6(1), 105-122. Informing Science Institute
- Mall, R. (2020). Fundamentals of software engineering (4th ed.). Retrieved from Eden, A. (2007). The three paradigms of computer science. Retrieved from https://eclass.hmu.gr/modules/document/file.php/TP374/Lecture%20Presentations/L1%20-%20 Introduction/Literature/2007%20-%20The%20 three%20paradigms%20of%20computer%20science%20-%20A.%20Eden.pdf

< About Me >



Educational Background

Most of my Elementary and Highschool days were spent attending Bright Minds in Action Learning Village before eventually we moved and I started attending Senior Highschool in UC

My Reason for Choosing IT

The reason for me choosing BSIT is because i've always had an interest in computers when I first got my own and I love to help people with technological issues so IT was a good choice for me.