Society, Ethics and Professionalism (SEP)

Preamble

While technical issues dominate the computing curriculum, they do not constitute a complete educational program in the broader context. Students must also be exposed to the larger societal context of computing to develop an understanding of the relevant social, ethical, legal and professional issues. This need to incorporate the study of these non-technical issues into the ACM curriculum was formally recognized in 1991, as can be seen from the following excerpt from CS1991 [1]:

> *Undergraduates also need to understand the basic cultural, social, legal, and ethical issues inherent in the discipline of computing. They should understand where the discipline has been, where it is, and where it is heading. They should also understand their individual roles in this process, as well as appreciate the philosophical questions, technical problems, and aesthetic values that play an important part in the development of the discipline.*
>
> *Students also need to develop the ability to ask serious questions about the social impact of computing and to evaluate proposed answers to those questions. Future practitioners must be able to anticipate the impact of introducing a given product into a given environment. Will that product enhance or degrade the quality of life? What will the impact be upon individuals, groups, and institutions?*
>
> *Finally, students need to be aware of the basic legal rights of software and hardware vendors and users, and they also need to appreciate the ethical values that are the basis for those rights. Future practitioners must understand the responsibility that they will bear, and the possible consequences of failure. They must understand their own limitations as well as the limitations of their tools. All practitioners must make a long-term commitment to remaining current in their chosen specialties and in the discipline of computing as a whole.*

As technological advances (more specifically, how these advances are used by humans) continue to significantly impact the way we live and work, the critical importance of social and ethical issues and professional practice continues to increase in importance and consequence. The ways humans use computer-based products and platforms, while hopefully providing opportunities, also introduce ever more challenging problems each year. A recent example is the emergence of Generative AI including large language models that generate code. A 2020 *Communications of the ACM* article [2] stated: “because computing as a discipline is becoming progressively more entangled within the human and social lifeworld, computing as an academic discipline must move away from engineering-inspired curricular models and integrate the analytic lenses supplied by social science theories and methodologies.”

In parallel to a heightened awareness of the social consequences computing has on the world, computing communities have become much more aware - and active - in areas of Inclusion, Diversity, Equity and Accessibility. All computing students deserve an inclusive, diverse, equitable and accessible inclusive learning environment. However, computing students have a unique duty to ensure that when put to practice, their skills, knowledge, and competencies are applied in similar fashion, ethically and professionally, in the society they are in. For these reasons, inclusion, diversity, equity and accessibility are inherently a part of Society, Ethics, and Professionalism, and a new knowledge unit has been added that addresses this.

Computer science educators may opt to deliver the material in this knowledge area integrated into the context of traditional technical and theoretical courses, in dedicated courses (ideally a combination of both) and as special units as part of capstone, project, and professional practice courses. <u>The material in this knowledge area is best covered through a combination of all the above</u>. It is too commonly held that many topics in knowledge units listed as CS Core may not readily lend themselves to being covered in other more traditional computer science courses. However many of these topics naturally arise and others can be included with minimal effort, and the benefits of exposing students to these topics within the context of those traditional courses is most valuable. Nonetheless institutional challenges will present barriers; for instance some of these traditional courses may not be offered at a given institution and in such cases it is difficult to cover these topics appropriately without a dedicated course. However, if social, ethical and professional considerations are covered only in a dedicated course and not in the context of others, it could reinforce the false notion that technical processes are void of these important aspects, or that they are more isolated than they are in reality. Because of the broad relevance of these knowledge units, it is important that as many traditional courses as possible include aspects such as case studies that analyze ethical, legal, social and professional considerations in the context of the technical subject matter of those courses. Courses in areas such as software engineering, databases, computer graphics, computer networks, information assurance and security, and introduction to computing provide obvious context for analysis of such issues. However, an ethics-related module could be developed for almost any course in the curriculum. It would be explicitly against the spirit of these recommendations to have only a dedicated course. Further, these topics should be covered in courses starting from year 1. Presenting them as advanced topics in later courses only creates an artificial perception that SEP topics are only important at a certain level or complexity. While it is true that the importance and consequence of SEP topics *increases* with level and complexity, introductory topics are not devoid of SEP topics. Further, many SEP topics are *best* presented early to lay a foundation for more intricate topics later in the curriculum.

Running through all the issues in this area is the need to speak to the computing practitioner’s responsibility to proactively address these issues by both ethical and technical actions. Today it is important not only for the topics in this knowledge area, but for students’ knowledge in general, that the ethical issues discussed in any course should be directly related to - and arise naturally from - the subject matter of that course. Examples include a discussion in a database course of the societal, ethical and professional aspects of data aggregation or data mining, or a discussion in a software engineering course of the potential conflicts between obligations to the customer and users as well as all others affected by their work. Computing faculty who are unfamiliar with the content and/or pedagogy of applied ethics are urged to take advantage of the considerable resources from ACM, IEEE-CS, SIGCAS (ACM Special Interest Group on Computers and Society), and other organizations. Additionally, it is the educator's responsibility to impress upon students that this area is just as important - in ways more important - than technical areas. The societal, ethical, and professional knowledge gained in studying topics in this knowledge area will be used throughout one's career and are transferable between projects, jobs, and even industries, particularly as one's career progresses into project leadership and management.

The ACM Code of Ethics and Professional Conduct [3], the IEEE Code of Ethics [4], and the AAAI Code of Ethics and Professional Conduct [5] provide guidance that serve as the basis for the conduct of all computing professionals in their work. The ACM Code emphasizes that ethical reasoning is not an algorithm to be followed and computer professionals are expected to consider how their work impacts the public good as the primary consideration. It falls to computing educators to highlight the domain-specific role of these topics for our students, but programs should certainly be willing to lean heavily on complementary courses from the humanities and social sciences.

We observe that computing educators are not moral philosophers. Yet CS2023, as with past CS curricular recommendations, indicate the need for ethical analysis. CS2023 along with all previous CS Curricular reports are quite clear on the required mathematical foundations that students are expected to gain and the courses from mathematics departments that provide such training. Yet, the same is not true of moral philosophy. No one would expect a student to be able to provide a proof by induction until after having successfully completed a course in discrete mathematics. Yet the parallel with respect to ethical analyses is somehow absent. We seemingly do expect our students to perform ethical analysis without having the appropriate prerequisite knowledge from philosophy.

The lack of such prerequisite training has facilitated graduates operating with a certain ethical egoism (e.g., 'Here's what I believe/think/feel is right'). However, regardless of how well intentioned, one might conclude that this is what brought us to this point in history where computer crimes, hacks, scandals, data breaches, and the general misuse of computing technology (including the data it consumes and produces) is a frequent occurrence. Certainly, computing graduates who have learned how to apply the various ethical frameworks or lenses proposed through the ages would only serve to improve this situation. In retrospect, to ignore the lessons from moral philosophy, which have been debated and refined for millenia, on what it means to act justly, or work for the common good, appears as hubris.

Changes since CS 2013

- Inclusion of SEP/Inclusion, Diversity, Equity, and Accessibility knowledge unit
- Changed titles of two knowledge units (e.g. Professional Communication -> Communication, Analytical Tools -> Methods for Ethical Analysis)

[1] ACM/IEEE-CS Joint Curriculum Task Force, Computing Curricula 1991 (1991), ACM Press and IEEE Computer Society Press.

[2] Randy Connolly. 2020. Why computing belongs within the social sciences. Commun. ACM 63, 8 (August 2020), 54–59. https://doi.org/10.1145/3383444

[3] ACM Code of Ethics and Professional Conduct. www.acm.org/about/code-of-ethics

[4] IEEE Code of Ethics on Professional Activities. https://www.ieee.org/about/corporate/governance/p7-8.html

[5] AAAI Code of Professional Ethics and Conduct. https://aaai.org/Conferences/code-of-ethics-and-conduct.php

Core Hours

Knowledge Units	**CS Core**	**KA Core**
Social Context	3	2
Methods for Ethical Analysis	2	1
Professional Ethics	2	2
Intellectual Property	1	1
Privacy and Civil Liberties	2	1
Communication	2	1
Sustainability	1	1
History	1	1
Economies of Computing	0	1
Security Policies, Laws and Computer Crimes	2	1
Equity, Diversity and Inclusion	2	2
Total	**18**	**14**

Knowledge Units

SEP-Context: Social Context

Computers, the internet, and artificial intelligence, perhaps more than any other technologies, have transformed society over the past several decades, with dramatic increases in human productivity; an explosion of options for news, entertainment, and communication; and fundamental breakthroughs in almost every branch of science and engineering. It is also imperative to recognize that this is not a one-way street. Society also affects computing, resulting in a complex socio-technical context that is constantly changing, requiring the perspective of history to put the present, as well as possible futures, into appropriate perspective.

Social Context provides the foundation for all other knowledge units in SEP, particularly Professional Ethics.

CS Core:

1. Social implications (e.g. political and cultural ideologies) in a hyper-networked world where the capabilities and impact of social media, artificial intelligence and computing in general are rapidly evolving
2. Impact of computing applications (e.g. social media, artificial intelligence applications) on individual well-being, and safety of all kinds (e.g., physical, emotional, economic)
3. Consequences of involving computing technologies, particularly artificial intelligence, biometric technologies and algorithmic decision-making systems, in civic life (e.g., facial recognition technology, biometric tags, resource distribution algorithms, policing software)
4. How deficits in diversity and accessibility in computing affect society and what steps can be taken to improve diversity and accessibility in computing

KA Core:

5. Growth and control of the internet, computing, and artificial intelligence
6. Often referred to as the digital divide, differences in access to digital technology resources and its resulting ramifications for gender, class, ethnicity, geography, and/or underdeveloped countries
7. Accessibility issues, including legal requirements and dark patterns
8. Context-aware computing

Illustrative Learning Outcomes:

CS Core:

1. Describe different ways that computer technology (networks, mobile computing, cloud computing) mediates social interaction at the personal and social group level.
2. Identify developers' assumptions and values embedded in hardware and software design, especially as they pertain to usability for diverse populations including under-represented populations and the disabled.
3. Interpret the social context of a given design and its implementation.
4. Evaluate the efficacy of a given design and implementation using empirical data.
5. Articulate the implications of social media use for different identities, cultures, and communities.

KA Core:

6. Explain the internet's role in facilitating communication between citizens, governments, and each other.
7. Analyze the effects of reliance on computing in the implementation of democracy (e.g., delivery of social services, electronic voting).
8. Describe the impact of the under-representation of people from historically minoritized populations in the computing profession (e.g., industry culture, product diversity).
9. Explain the implications of context awareness in ubiquitous computing systems.
10. Explain how access to the internet and computing technologies affect different societies.
11. Discuss why/how internet access can be viewed as a human right.

SEP-Ethical-Analysis: Methods for Ethical Analysis

Ethical theories and principles are the foundations of ethical analysis because they are the viewpoints which can provide guidance along the pathway to a decision. Each theory emphasizes different assumptions and methods for determining the ethicality of a given action. It is important for students to recognize that decisions in different contexts may require different ethical theories to arrive at ethically acceptable outcomes, and what constitutes 'acceptable' depends on a variety of factors such as cultural context. Applying methods for ethical analysis requires both an understanding of the underlying principles and assumptions guiding a given tool and an awareness of the social context for that decision. Traditional ethical frameworks as provided by western philosophy can be useful, but they are not all-inclusive. Effort must be taken to include decolonial, indigenous and historically marginalized ethical perspectives whenever possible. No theory will be universally applicable to all contexts, nor is any single ethical framework the 'best'. Engagement across various ethical schools of thought is important for students to develop the critical thinking needed in judiciously applying methods for ethical analysis of a given situation.

CS Core:

1. Avoiding fallacies and misrepresentation in argumentation
2. Ethical theories and decision-making (philosophical and social frameworks)
3. Recognition of the role culture plays in our understanding, adoption, design, and use of computing technology
4. Why ethics is important in computing, and how ethics is similar to, and different from, laws and social norms

KA Core:

5. Professional checklists
6. Evaluation rubrics
7. Stakeholder analysis
8. Standpoint theory
9. Introduction to ethical frameworks (e.g., consequentialism such as utilitarianism, non-consequentialism such as duty, rights or justice, agent-centered such as virtue or feminism, contractarianism, ethics of care) and their use for analyzing an ethical dilemma

Illustrative Learning Outcomes:

CS Core:

1. Recognize and describe how a given cultural context impacts decision making.
2. Illustrate the use of example and analogy in ethical argument.
3. Analyze (and avoid) basic logical fallacies in an argument.
4. Analyze an argument to identify premises and conclusion.
5. Evaluate how and why ethics is so important in computing and how it relates to cultural norms, values, and law.
6. Justify a decision made on ethical grounds.

KA Core:

7. Evaluate all stakeholder positions in relation to their cultural context in a given situation.
8. Evaluate the potential for introducing or perpetuating ethical debt (deferred consideration of ethical impacts or implications) in technical decisions.
9. Discuss the advantages and disadvantages of traditional ethical frameworks

10. Analyze ethical dilemmas related to the creation and use of technology from multiple perspectives using ethical frameworks

SEP-Professional-Ethics: Professional Ethics

Computer ethics is a branch of practical philosophy that deals with how computing professionals should make decisions regarding professional and social conduct. There are three primary influences: 1) The individual's own personal ethical code, 2) Any informal or formal code of ethical behavior existing in the workplace, applicable licensures or certifications, and 3) Exposure to formal codes of ethics and ethical frameworks.

CS Core:

1. Community values and the laws by which we live
2. The nature of professionalism including care, attention and discipline, fiduciary responsibility, and mentoring
3. Keeping up-to-date as a computing professional in terms of familiarity, tools, skills, legal and professional frameworks as well as the ability to self-assess and progress in the computing field
4. Professional certification, codes of ethics, conduct, and practice, such as the ACM/IEEE-CS, SE, AITP, IFIP and international societies
5. Accountability, responsibility and liability (e.g., software correctness, reliability and safety, warranty, negligence, strict liability, ethical approaches to security vulnerability disclosures)
6. Introduction to theories describing the human creation and use of technology including instrumentalism, sociology of technological systems, disability justice, neutrality thesis, pragmatism, and decolonial theories
7. Strategies for recognizing and reporting designs, systems, software, and professional conduct (or their outcomes) that may violate law or professional codes of ethics.

KA Core:

8. The role of the computing professional and professional societies in public policy
9. Maintaining awareness of consequences
10. Ethical dissent and whistle-blowing
11. The relationship between regional culture and ethical dilemmas
12. Dealing with harassment and discrimination
13. Forms of professional credentialing
14. Ergonomics and healthy computing environments
15. Time to market and cost considerations versus quality professional standards

Illustrative Learning Outcomes:

CS Core:

1. Identify ethical issues that arise in software design, development practices, and software deployment
2. Demonstrate how to address ethical issues in specific situations.
3. Explain the ethical responsibility of ensuring software correctness, reliability and safety including from where this responsibility arises (e.g., ACM/IEEE/AAAI Codes of Ethics, laws and regulations, organizational policies).
4. Describe the mechanisms that typically exist for a professional to keep up-to-date in ethical matters.
5. Describe the strengths and weaknesses of relevant professional codes as expressions of professionalism and guides to decision-making.
6. Analyze a global computing issue, observing the role of professionals and government officials in managing this problem.

KA Core:

8. Describe ways in which professionals and professional organizations may contribute to public policy.
9. Describe the consequences of inappropriate professional behavior.
10. Be familiar with whistleblowing and have access to knowledge to guide one through an incident.
11. Provide examples of how regional culture interplays with ethical dilemmas.
12. Discuss forms of harassment and discrimination and avenues of assistance.
13. Examine various forms of professional credentialing.
14. Explain the relationship between ergonomics in computing environments and people's health.
15. Describe issues associated with industries' push to focus on time to market versus enforcing quality professional standards.

SEP-IP: Intellectual Property

Intellectual property refers to a range of intangible rights of ownership in any product of the human intellect, such as a software program. Laws, which vary by country, provide different methods for protecting these rights of ownership based on their type. Ideally, intellectual property laws balance the interests of creators and of users of the property. There are essentially four types of intellectual property rights relevant to software: patents, copyrights, trade secrets and trademarks. Moreover, property rights are also often protected by user licenses. Each affords a different type of legal protection.

CS Core:

1. Intellectual property rights
2. Intangible digital intellectual property (IDIP)
3. Legal foundations for intellectual property protection
4. Common software licenses (e.g., MIT, GPL and its variants, Apache, Mozilla, Creative Commons)
5. Plagiarism and authorship

KA Core:

6. Philosophical foundations of intellectual property
7. Forms of intellectual property (e.g., copyrights, patents, trade secrets, trademarks) and the rights they protect
8. Limitations on copyright protections, including fair use and the first sale doctrine
9. Intellectual property laws and treaties that impact the enforcement of copyrights
10. Software piracy and technical methods for enforcing intellectual property rights, such as digital rights management and closed source software as a trade secret
11. Moral and legal foundations of the open source movement
12. Systems that use others' data (e.g., large language models)

Illustrative Learning Outcomes:

CS Core:

1. Describe and critique legislation and precedent aimed at digital copyright infringements.
2. Identify contemporary examples of intangible digital intellectual property.
3. Select an appropriate software license for a given project.
4. Justify legal and ethical uses of copyrighted materials.

5. Interpret the intent and implementation of software licensing.
6. Determine whether a use of copyrighted material is likely to be fair use.
7. Evaluate the ethical issues inherent in various plagiarism detection mechanisms.
8. Identify multiple forms of plagiarism beyond verbatim copying of text or software (e.g., intentional paraphrasing, authorship misrepresentation, and improper attribution).

KA Core:

9. Discuss the philosophical bases of intellectual property in an appropriate context (e.g., country, etc.).
10. Weigh the conflicting issues involved in securing software patents.
11. Characterize and contrast the protections and obligations of copyright, patent, trade secret, and trademarks.
12. Explain the rationale for the legal protection of intellectual property in the appropriate context (e.g., country, etc.).
13. Evaluate the use of copyrighted work under the concepts of fair use and the first sale doctrine.
14. Identify the goals of the open source movement and its impact on fields beyond computing, such as the right-to-repair movement.
15. Characterize the global nature of software piracy.
16. Critique the use of technical measures of digital rights management (e.g., encryption, watermarking, copy restrictions, and region lockouts) from multiple stakeholder perspectives.
17. Discuss the nature of anti-circumvention laws in the context of copyright protection.

SEP-Privacy: Privacy and Civil Liberties

Electronic information sharing highlights the need to balance privacy protections with information access. The ease of digital access to many types of data makes privacy rights and civil liberties more complex, especially given cultural and legal differences in these areas. Complicating matters further, privacy also has interpersonal, organizational, business, and governmental components. In addition, the interconnected nature of online communities raises challenges for managing expectations and protections for freedom of expression in various cultures and nations. Technology companies that provide platforms for user-generated content are under increasing pressure to perform governance tasks, potentially facing liability for their decisions.

CS Core:

1. Privacy implications of widespread data collection including but not limited to transactional databases, data warehouses, surveillance systems, and cloud computing
2. Conceptions of anonymity, pseudonymity, and identity
3. Technology-based solutions for privacy protection (e.g., end-to-end encryption and differential privacy)
4. Civil liberties and cultural differences

KA Core:

5. Philosophical and legal conceptions of the nature of privacy
6. Legal foundations of privacy protection in relevant jurisdictions (e.g., GDPR in the EU)
7. Privacy legislation in areas of practice (e.g., HIPAA in the US)
8. Freedom of expression and its limitations
9. User-generated content, content moderation, and liability

Illustrative Learning Outcomes:

CS Core:

1. Evaluate solutions to privacy threats in transactional databases and data warehouses.
2. Describe the role of data collection in the implementation of pervasive surveillance systems (e.g., RFID, face recognition, toll collection, mobile computing).
3. Distinguish the concepts and goals of anonymity and pseudonymity
4. Describe the ramifications of technology-based privacy protections, including differential privacy and end-to-end encryption
5. Identify cultural differences regarding the nature and necessity of privacy and other civil liberties.

KA Core:

6. Discuss the philosophical basis for the legal protection of personal privacy in an appropriate context (e.g., country, etc.).
7. Critique the intent, potential value and implementation of various forms of privacy legislation.
8. Identify strategies to enable appropriate freedom of expression.

SEP-Communication

Computing is an inherently collaborative and social discipline making communication an essential aspect of computing. Much but not all of this communication occurs in a professional setting where communication styles, expectations, and norms differ from other contexts where similar technology, such as email or messaging, might be used. Both professional and informal communication conveys information to various audiences who may have very different goals and needs for that information. It is also important to note that computing professionals are not just communicators, but are also listeners who must be able to hear and thoughtfully make use of feedback received from various stakeholders. Effective communication skills are not something one 'just knows' - they are developed and can be learned. Communication skills are best taught in context throughout the undergraduate curriculum.

CS Core:

1. Interpreting, summarizing, and synthesizing technical material, including source code and documentation
2. Writing effective technical documentation and materials (tutorials, reference materials, API documentation)
3. Identifying, describing, and employing (clear, polite, concise) oral, written, and electronic team and group communication.
4. Understanding and enacting awareness of audience in communication by communicating effectively with different stakeholders such as customers, leadership, or the general public
5. Appropriate and effective team communication including utilizing collaboration tools and conflict resolution
6. Recognizing and avoiding the use of rhetorical fallacies when resolving technical disputes
7. Understanding accessibility and inclusivity requirements for addressing professional audiences

KA Core:

8. Demonstrating cultural competence in written and verbal communication

9. Using synthesis to concisely and accurately convey tradeoffs in competing values driving software projects including technology, structure/process, quality, people, market and financial
10. Use writing to solve problems or make recommendations in the workplace, such as raising ethical concerns or addressing accessibility issues

Illustrative Learning Outcomes:

CS Core:

1. Understand the importance of writing concise and accurate technical documents following well-defined standards for format and for including appropriate tables, figures, and references.
2. Evaluate written technical documentation for technical accuracy, concision, lack of ambiguity, and awareness of audience.
3. Develop and deliver an audience aware, accessible, and organized formal presentation.
4. Plan interactions (e.g., virtual, face-to-face, shared documents) with others in ways that invite inclusive participation, model respectful consideration of others' contributions, and explicitly value diversity of ideas.
5. Recognize and describe qualities of effective communication (e.g., virtual, face-to-face, intragroup, shared documents).
6. Understand how to effectively and appropriately communicate as a member of a team including conflict resolution techniques.

KA Core:

7. Discuss ways to influence performance and results in diverse and cross-cultural teams.
8. Evaluate personal strengths and weaknesses to work remotely as part of a team drawing from diverse backgrounds and experiences.

SEP-Sustainability

Sustainability is defined by the United Nations as "development that meets the needs of the present without compromising the ability of future generations to meet their own needs."[1] Alternatively, it is the "balance between the environment, equity and economy."[2] As computing extends into more and more aspects of human existence, we are already seeing estimates that 10% of global electricity usage is spent on computing, and that percentage will continue growing. Further, electronics contribute individually to demand for rare earth elements, mineral extraction, and countless e-waste concerns. Students should be prepared to engage with computing with a background that recognizes these global and environmental costs and their potential long term effects on the environment and local communities.

[1] https://www.un.org/en/academic-impact/sustainability

[2] https://www.sustain.ucla.edu/what-is-sustainability

CS Core:

1. Being a sustainable practitioner by taking into consideration environmental, social, and cultural impacts of implementation decisions (e.g., sustainability goals, algorithmic bias/outcomes, economic viability, and resource consumption)

2. Local/regional/global social and environmental impacts of computing systems use and disposal (e.g. carbon footprints, e-waste) in hardware (e.g., data centers) and software (e.g. blockchain, AI model training and use).
3. Discuss the tradeoffs involved in proof-of-work and proof-of-stake algorithms

KA Core:

4. Guidelines for sustainable design standards
5. Systemic effects of complex computer-mediated phenomena (e.g., social media, offshoring, remote work)
6. Pervasive computing: Information processing that has been integrated into everyday objects and activities, such as smart energy systems, social networking and feedback systems to promote sustainable behavior, transportation, environmental monitoring, citizen science and activism
7. Conduct research on applications of computing to environmental issues, such as energy, pollution, resource usage, recycling and reuse, food management / production, and others
8. How the sustainability of software systems are interdependent with social systems, including the knowledge and skills of its users, organizational processes and policies, and its societal context (e.g., market forces, government policies)

Illustrative Learning Outcomes:

CS Core:

1. Identify ways to be a sustainable practitioner.
2. For any given project (software artifact, hardware, etc.) evaluate the environmental impacts of its deployment. (e.g., energy consumption, contribution to e-waste, impact of manufacturing).
3. Illustrate global social and environmental impacts of computer use and disposal (e-waste).
4. List the sustainable effects of modern practices and activities (e.g., remote work, online commerce, cryptocurrencies, data centers).

KA Core:

5. Describe the environmental impacts of design choices within the field of computing that relate to algorithm design, operating system design, networking design, database design, etc.
6. Investigate the social and environmental impacts of new system designs.
7. Identify guidelines for sustainable IT design or deployment.
8. Investigate pervasive computing in areas such as smart energy systems, social networking, transportation, agriculture, supply-chain systems, environmental monitoring and citizen activism.
9. Assess computing applications in respect to environmental issues (e.g., energy, pollution, resource usage, recycling and reuse, food management and production).

SEP-History

History is important because it provides a mechanism for understanding why our computing systems operate the way they do, the societal contexts in which these approaches arose, and how those continue to echo through the discipline today. This history of computing is taught to provide a sense of how the rapid change in computing impacts society on a global scale. It is often taught in context with foundational concepts, such as system fundamentals and software development fundamentals.

CS Core:

1. The history of computing: hardware, software, and human/organizational and the role of this in present social contexts

KA Core:

2. Age I: Prehistory—the world before ENIAC (1946): Ancient analog computing (Stonehenge, Antikythera mechanism, Salisbury Cathedral clock, etc.), human-calculated number tables, Euclid, Lovelace, Babbage, Gödel, Church, Turing, pre-electronic (electro-mechanical and mechanical) hardware
3. Age II: Early modern (digital) computing - ENIAC, UNIVAC, Bombes (Bletchley Park codebreakers), computer companies (e.g., IBM), mainframes, etc.
4. Age III: Modern (digital) computing - PCs, modern computer hardware and software, Moore's Law
5. Age IV: Internet - networking, internet architecture, browsers and their evolution, standards, big players (Google, Amazon, Microsoft, etc.), distributed computing
6. Age V: Cloud - smartphones (Apple, Android, and minor ones), cloud computing, remote servers, software as a service (SaaS), security and privacy, social media
7. Age VI: Emerging AI-assisted technologies including decision making systems, recommendation systems, generative AI and other machine learning driven tools and technologies

Illustrative Learning Outcomes:

CS Core:

1. Understand the relevance and impact of computing history on recent events, present context, and possible future outcomes. *Ideally from more than one cultural perspective.*

KA Core:

2. Identify significant trends in the history of the computing field.
3. Identify the contributions of several pioneering individuals or organizations (research labs, computer companies, government offices) in the computing field.
4. Discuss the historical context for important moments in history of computing, such as the move from vacuum tubes to transistors (TRADIC), early seminal operating systems (e.g., OS 360), Xerox PARC and the first Apple computer with a GUI, the creation of specific programming language paradigms, the first computer virus, the creation of the internet, the creation of the WWW, the dot com bust, Y2K, the introduction of smartphones, etc.
5. Compare daily life before and after the advent of personal computers and the Internet.

SEP-Economies: Economies of Computing

The economies of computing are important to those who develop and provide computing resources and services to others as well as society in general. They are equally important to users of these resources and services, both professional and non-professional.

KA Core:

1. Economies of providers: regulated and unregulated, monopolies, network effects, and open-market. “Walled Gardens” in tech environments
2. The knowledge and attention economies
3. Effect of skilled labor supply and demand on the quality of computing products
4. Pricing strategies in the computing domain: subscriptions, planned obsolescence, software licenses, open-source, free software
5. Outsourcing and off-shoring software development; impacts on employment and on economics
6. Consequences of globalization for the computer science profession and users
7. Differences in access to computing resources and the possible effects thereof
8. Automation and its effect on job markets, developers, and users
9. Economies of scale, startups, entrepreneurship, philanthropy
10. How computing is changing personal finance: Blockchain and cryptocurrencies, mobile banking and payments, SMS payment in developing regions, etc.

Illustrative Learning Outcomes:

KA Core:

1. Summarize concerns about monopolies in tech, walled gardens vs open environments, etc.
2. Identify several ways in which the information technology industry and users are affected by shortages in the labor supply.
3. Outline the evolution of pricing strategies for computing goods and services.
4. Explain the social effects of the knowledge and attention economies.
5. Summarize the consequences of globalization and nationalism in the computing industry.
6. Describe the effects of automation on society, and job markets in particular.
7. Detail how computing has changed the corporate landscape
8. Outline how computing has changed personal finance and the consequences of this, both positive and negative.

SEP-Security: Security Policies, Laws and Computer Crimes

While security policies, laws and computer crimes are important, it is essential they are viewed with the foundation of other Social and Professional knowledge units, such as Intellectual Property, Privacy and Civil Liberties, Social Context, and Professional Ethics. Computers, the internet, and artificial intelligence, perhaps more than any other technologies, have transformed society over the past 75 years. At the same time, they have contributed to unprecedented threats to privacy; new categories of computer crime and anti-social behavior; major disruptions to organizations; and the large-scale concentration of risk into information systems.

CS Core:

1. Examples of computer crimes and legal redress for computer criminals
2. Social engineering, computing-enabled fraud, and recovery
3. Identify what constitutes computer crime, such as Issues surrounding the misuse of access and breaches in security
4. Motivations and ramifications of cyber terrorism and criminal hacking, “cracking”

5. Effects of malware, such as viruses, worms and Trojans
6. Attacks on critical infrastructure such as electrical grids and pipelines

KA Core:

7. Benefits and challenges of existing and proposed computer crime laws
8. Security policies and the challenges of compliance
9. Responsibility for security throughout the computing life cycle
10. International and local laws and how they intersect

Illustrative Learning Outcomes:

CS Core:

1. List classic examples of computer crimes and social engineering incidents with societal impact.
2. Identify laws that apply to computer crimes.
3. Describe the motivation and ramifications of cyber terrorism, data theft, hacktivism (hacking as activism), ransomware, and other attacks..
4. Examine the ethical and legal issues surrounding the misuse of access and various breaches in security.
5. Discuss the professional's role in security and the trade-offs involved.

KA Core:

6. Investigate measures that can be taken by both individuals and organizations including governments to prevent or mitigate the undesirable effects of computer crimes and identity theft.
7. Draft a company-wide security policy, which includes procedures for managing passwords and employee monitoring.
8. Understand how legislation from one region may affect activities in another (e.g. how EU GDPR applies globally, when EU persons are involved).

SEP-IDEA: Inclusion, Diversity, Equity, and Accessibility

Computer Science has had—since its inception as a field—a diversity problem. Despite being a creative, highly compensated field with myriad job (and other) opportunities, racial, gender and other inequities in representation are pervasive. For too many students, their first computer science course is their last. There are many factors including the legacy of systemic racism, ableism, sexism, classism, and other injustices that contribute to the lack of diverse identities within computer science, and there is no single, quick fix.

CS2023's sponsoring organizations are ACM, IEEE CS, and AAAI. Each of those organizations [https://www.acm.org/diversity-inclusion/about#DEIPrinciples, https://www.ieee.org/about/diversity-index.html, https://aaai.org/Organization/diversity-statement.php] place a high value on inclusion, diversity, equity, and accessibility; and our computer science classrooms should promote and model those principles. We should welcome and seek diversity—the gamut of human differences including gender, gender identity, race, politics, ability and attributes, religion, nationality, etc.—in our classrooms, departments and campuses. We should strive to make our classrooms, labs, and curricula accessible and to promote inclusion; the sense of belonging we feel in a community where we are respected and wanted. To achieve equity, we must allocate resources, promote fairness, and check our biases to ensure persons of all identities achieve success. Accessibility should be considered and implemented in all computing activities and products.

Explicitly infusing inclusion, diversity, equity, and accessibility across the computer science curriculum demonstrates its importance for the department, institution, and our field—all of which likely have a IDEA statement and/or initiative(s). This emphasis on IDEA is important ethically and a bellwether issue of our time. Many professionals in computing already recognize attention to diversity, equity, inclusion, accessibility as integral parts of disciplinary practice. Regardless of the degree to which IDEA values appear in any one computer science class, research suggests that a lack of attention to IDEA will result in inferior designs. Not only does data support that diverse teams outperform homogeneous ones, but diverse teams may have prevented egregious technology failures in the headlines such as facial recognition misuse, airbag injuries, and deaths.

CS Core:

1. How identity impacts and is impacted by computing environments (academic and professional) and technologies
2. The benefits of diverse development teams and the impacts of teams that are not diverse.
3. Inclusive language and charged terminology, and why their use matters
4. Inclusive behaviors and why they matter
5. Designing and developing technology with accessibility in mind
6. Designing for accessibility
7. How computing professionals can influence and impact inclusion, diversity, equity, and accessibility both positively and negatively, not only through the software they create.

KA Core:

8. Highlight experts (practitioners, graduates, and upper level students) who reflect the identities of the classroom and the world
9. Benefits of diversity and harms caused by a lack of diversity
10. Historic marginalization due to technological supremacy and global infrastructure challenges to equity and accessibility

Illustrative Learning Outcomes:

CS Core:

1. Define and distinguish equity, equality, diversity, inclusion, and accessibility..
2. Describe the impact of power and privilege in the computing profession as it relates to culture, industry, products, and society.
3. Identify language, practices, and behaviors that may make someone feel included in a workplace and/or a team, and why is it relevant. Avoid charged terminology - see *Words Matter* (https://www.acm.org/diversity-inclusion/words-matter).
4. Evaluate the accessibility of your classroom or lab. Evaluate the accessibility of your webpage. (See https://www.w3.org/WAI/.)
5. Work collegially and respectfully with team members who do not share your identity. *It is not enough to merely assign team projects. Faculty should prepare students for*

teamwork and monitor, mentor, and assess the effectiveness of their student teams throughout a project.

6. Compare the demographics of your institution's computer science and STEM majors to the overall institutional demographics. If they differ, identify factors that contribute to inequitable access, engagement, and achievement in computer science among marginalized groups.
7. Compare the demographics of your institution to the overall community demographics. If they differ, identify factors that contribute to inequitable access, engagement, and achievement among marginalized groups.
8. Identify developers' assumptions and values embedded in hardware and software design, especially as they pertain to usability by diverse populations.

KA Core:

9. Highlight experts (practitioners, graduates, and upper level students—current and historic) who reflect the identities of the classroom and the world.
10. Identify examples of the benefits that diverse teams can bring to software products, and those where a lack of diversity have costs.
11. Give examples of systemic changes that could positively address diversity, equity, and inclusion in a familiar context (i.e. in an introductory computing course).

Professional Dispositions

- Critical Self-reflection - Being able to inspect one's own actions, thoughts, biases, privileges, and motives will help in discovering places where professional activity is not up to current standards. Understand both conscious and unconscious bias and continuously work to counteract them.
- Responsiveness - Ability to quickly and accurately respond to changes in the field and adapt in a professional manner, such as shifting from in-person office work to remote work at home. These shifts require us to rethink our entire approach to what is considered "professional".
- Proactiveness - Being professional in the workplace means finding new trends (e.g. in accessibility or inclusion) and understanding how to implement them immediately for a more professional working environment.
- Cultural Competence - Prioritize cultural competence—the ability to work with people from cultures different from your own—by using inclusive language, watching for and counteracting conscious and unconscious bias, and encouraging honest and open communication.
- Advocation - Thinking, speaking and acting in ways that foster and promote inclusion, diversity, equity and accessibility in all ways including but not limited to teamwork, communication, and developing products (hardware and software).

Course Packaging Suggestions

In computing, Societal, Ethical, and Professional topics arise in all other knowledge areas and therefore should arise in the context of other computing courses, not just siloed in an "SEP course". These topics should be covered in courses starting from year 1(the only likely exception is SEP-Ethical-Analysis: Methods for Ethical Analysis) which could be delivered as part of a first-year course or via a seminar or an online independent study.

Presenting SEP topics as advanced topics only covered in later courses could create the incorrect perception that SEP topics are only important at a certain level or complexity. While it is true that the importance and consequence of SEP topics *increases* with level and complexity, introductory topics are not devoid of SEP topics. Further, many SEP topics are *best* presented early to lay a foundation for more intricate topics later in the curriculum.

Who should teach some of these topics is a complex topic. When SEP topics arise in other courses these are naturally often taught by the instructor teaching that course, although at times bringing in expert educators from other disciplines (e.g., law, ethics, etc.) could be advantageous. Stand-alone courses in SEP could be taught by a team of CS and other disciplines - although more logistically complicated, this may be a better approach than being taught by a single CS instructor. Regardless, who teaches SEP topics and/or courses warrants

At a minimum the SEP CS Core learning outcomes* are best covered in the context of courses covering other knowledge areas - ideally the SEP KA Core hours are also.
***With the likely exception of SEP-Ethical Analysis: Methods for Ethical Analysis which could be delivered as discussed above.

At some institutions an **in-depth dedicated course** at the mid- or advanced-level may be offered covering all recommended topics in both the CS Core and KA Core knowledge units in close coordination with learning outcomes best covered in the context of courses covering other knowledge areas. Such a course could include:

- SEP-Context (5 hours)
- SEP-Ethical-Analysis: Methods for Ethical Analysis (3 hours)
- SEP-Professional-Ethics: Professional Ethics (4 hours)
- SEP-IP (2 hours)
- SEP-Privacy: Privacy and Civil Liberties (3 hours)
- SEP-Communication (3 hours)
- SEP-Sustainability (2 hours)
- SEP-History (2 hours)
- SEP-Economies: Economies of Computing (1 hour)
- SEP-Security: Security Policies, Laws and Computer Crimes (3 hours)
- SEP-IDEA: Diversity, Equity, and Inclusion (4 hours)

At some institutions a **dedicated minimal course** may be offered covering the CS Core knowledge units in close coordination with learning outcomes best covered in the context of courses covering other knowledge areas. Such a course could include:

- SEP-Context (3 hours)
- SEP-Ethical-Analysis: Methods for Ethical Analysis (2 hours)
- SEP-Professional-Ethics (2 hours)
- SEP-IP (1 hour)
- SEP-Privacy: Privacy and Civil Liberties (2 hours)
- SEP-Communication (2 hours)
- SEP-Sustainability (1 hour)
- SEP-History (1 hour)
- SEP-Security: Security Policies, Laws and Computer Crimes (2 hours)
- SEP-IDEA: Inclusion, Diversity, Equity, and Accessibility (2 hours)

References

1. Emanuelle Burton, Judy Goldsmith, Nicholas Mattei, Cory Siler, and Sara-Jo Swiatek. 2023. Teaching Computer Science Ethics Using Science Fiction. In Proceedings of the 54th ACM Technical Symposium on Computer Science Education

V. 2 (SIGCSE 2023). Association for Computing Machinery, New York, NY, USA, 1184. https://doi.org/10.1145/3545947.3569618
2. Randy Connolly. 2020. Why computing belongs within the social sciences. Commun. ACM 63, 8 (August 2020), 54–59. https://doi.org/10.1145/3383444
3. Casey Fiesler. Tech Ethics Curricula: A Collection of Syllabi Used to Teach Ethics in Technology Across Many Universities
    a. https://cfiesler.medium.com/tech-ethics-curricula-a-collection-of-syllabi-3eedfb76be18
    b. Tech Ethics Curricula
4. Casey Fiesler. Tech Ethics Readings: A Spreadsheet of Readings Used to Teach Ethics in Technology Tech Ethics Class Readings
5. Stanford Embedded EthiCS, Embedding Ethics in Computer Science. https://embeddedethics.stanford.edu/
6. Jeremy, Weinstein, Rob Reich, and Mehran Sahami. *System Error: Where Big Tech Went Wrong and How We Can Reboot*. Hodder Paperbacks, 2023.
7. Baecker, R. Computers in Society: Modern Perspectives, Oxford University Press. (2019).
8. Embedded EthiCS @ Harvard: bringing ethical reasoning into the computer science curriculum. https://embeddedethics.seas.harvard.edu/about