Teach AI CSTA

Guidance on the Future of Computer Science Education in an Age of AI

teachai.org/cs

Partners

Guidance on the Future of Computer Science Education in an Age of AI was developed by TeachAI and the Computer Science Teachers Association (CSTA) in partnership with a community of thought leaders.

Lead Partners

In partnership with Karen Brennan, Maya Israel, Shuchi Grover and Matti Tedre

Partners

TeachAI Steering Committee

In collaboration with the World Economic Forum


Break it Down
when rock hits
Rock hits blue

Table of Contents

1 **Executive Summary**

5 **Why is it Still Important to Learn to Program?**

11 **How Are Computer Science Educators Teaching With and About AI?**

19 **How Can Students Become Critical Consumers and Responsible Creators of AI?**

27 **The Role of AI in Computer Science Education: Results from a Teacher Survey**

Executive Summary

Recent innovations in artificial intelligence (AI) are provoking a wide range of questions about the powerful technology's role in the computer science (CS) classroom. Educators seeking to understand the opportunities and risks of AI are asking:

- Why is it still necessary to learn to program?
- What do students need to learn about AI?
- How do we leverage AI to broaden access and participation in CS?

While AI offers possibilities for supporting teachers' work and enhancing student learning, there are also concerns about disparate societal impacts. Teachers are concerned about student privacy, the overreliance on AI tools, and students not developing foundational CS skills. These risks demand a cautious approach by everyone in the education community and a commitment to ensuring that AI benefits all students. **Guidance on the Future of Computer Science Education in an Age of AI** is meant to inspire the community to reflect on what it means to teach with and about AI in CS education.

The Guidance

At a time when the entire education community is grappling with how to realize the benefits of AI while mitigating the risks, the briefs that compose **Guidance on the Future of Computer Science Education in an Age of AI** serve as the beginning of a discussion rather than definitive answers. The briefs offer preliminary insights for responsibly and effectively integrating AI into primary and secondary CS education, address common misconceptions, and provide a balanced perspective on critical issues.

Each brief culminates in reflection questions, audience-specific calls to action, and questions for further study. The guidance illustrates an education community grappling with what AI means for the future of their discipline and serves as an example for other disciplines.

The guidance can be used to support a broad audience:

- Teachers
- Administrators
- Curriculum Providers
- Professional Development Providers
- Standards Writers
- Edtech Developers
- Researchers

The Process

Developed by TeachAI and the Computer Science Teachers Association (CSTA), **Guidance on the Future of Computer Science Education in an Age of AI** was informed by:

- 72 TeachAI advisory committee and government agency participants,
- 46 expert interviews,
- 8 focus groups featuring over 100 teachers, researchers, and curriculum and professional development providers,
- a lead group of researchers and organizations, and
- a global working group of non-profit organizations, teacher associations, and industry leaders.

The guidance also draws on a literature review of dozens of research articles and the contributions of more than 360 educators worldwide who responded to our May 2024 CSTA/TeachAI survey. See the **Research Collection** at teachai.org/cs.

Each Brief

The first set of briefs in the guidance addresses three of the most pressing questions in CS education today:

1. **Why is it Still Important to Learn to Program?**
   Despite their coding abilities, AI tools are far from perfect, and learning to program lays the foundation for students to develop the conceptual understanding, agency, and dispositions needed to understand, use, and evaluate these tools and their outputs.

2. **How Are Computer Science Educators Teaching With and About AI?**
   Learning CS is a natural opportunity for students to explore the benefits, limitations, and societal impacts of emerging technologies. Early initiatives suggest that teaching with and about AI in CS education has the potential to promote computational thinking, demystify AI, and equip students to use AI tools responsibly. A mindful approach to implementing AI tools is critical to realizing the potential benefits of AI while mitigating the risks.

3. **How Can Students Become Critical Consumers and Responsible Creators of AI?**
   Many teachers have long advocated for teaching about technology's ethical and societal impacts as a core component of CS education. Amid the rise of Generative AI (Gen AI), CS teachers can help students develop the practices needed to become critical consumers and responsible creators of AI.

Future briefs in the series will consider questions such as:

- How can AI be used to broaden participation in CS?
- What core CS concepts and practices might be emphasized or deemphasized when we teach with and about AI?
- How can we better support students with disabilities to learn with and about AI?
- What can CS education learn from how AI is used in software development?
- What AI-related skills and content topics should a fundamental CS experience include?
- How can CS teachers use AI to promote creativity, starting in primary grades?
- How do we teach about AI in contexts with low technical infrastructure?

Additional AI in Education Resources

For those school systems ready to begin developing guidance on the responsible use of AI, the **AI Guidance for Schools Toolkit** provides seven principles for realizing the benefits of AI while mitigating the risks.

For those just starting to learn about AI, see **Foundational Policy Ideas for AI in Education** for resources designed for education leaders: **What Is AI?**, **AI in Education and the Workforce**, and **Classroom Perspectives on AI**.

Acknowledgments

We thank those who contributed their time and expertise to inform this guidance and look forward to continuing to learn together what it means to teach with and about AI in CS education.

This guidance was developed by TeachAI and the CSTA in partnership with Karen Brennan (Harvard University), Quinn Burke (Digital Promise), Marie desJardins (AAAI), Bruce Fuda (Grok Academy), Christina Gardner-McCune (AI4K12 and University of Florida), Shuchi Grover (Looking Glass Ventures and Edfinity), Maya Israel (University of Florida), Irene Lee (Everyday AI), Matti Tedre (University of Eastern Finland), and Jan Vahrenhold (German Informatics Society).

We thank the TeachAI Future of CS Workgroup, which includes Adobe, Amazon, Black in AI, College Board, Data Science 4 Everyone, Digital Promise, Education Above All, ETS, Texas Advanced Computing Center, GitHub, Google, Grok Academy, Infosys Foundation USA, Institute for Advancing Computing Education, ISTE, Microsoft, National Council of Teachers of Mathematics, NCWIT, RobinCode, and participating U.S. and global government agencies.

We'd also like to recognize Allen Antoine, Owen Astrachan, Elaine Atherton, Carla Brodley, Center for Computational Thinking and Design at Aarhus University, Edie Cheng, CS Access, Leigh Ann DeLyser, Charlotte Dungan, Crystal Furman, Dan Garcia, Kinnis Gosha, Mark Guzdial, William G. Harris, Sallie Holloway, Lori Jacques, Antti Kiviniemi, John Kleeman, Jill Kowalchuk, Michael Littman, Yolanda Lozano, Don Miller, John Mitchell, Rusty Nye, Jamie Payton, Chris Piech, Jennifer Rosato, Mehran Sahami, Vicky Sedgwick, Ben Shapiro, Sepehr Vakil, Sara Vogel, Hannah Walden, Shira Wein, Benji Xie, and Aman Yadav for their input.

TeachAI is led by a steering committee of Code.org, ETS, ISTE, Khan Academy, and the World Economic Forum.

Authors:
Pat Yongpradit, Code.org
Caron Carlson, Code.org
Veronica Ellis, Code.org
Bryan Twarek, CSTA
Jacob Koressel, CSTA

Suggested Citation
TeachAI & CSTA (2024). Guidance on the Future of Computer Science Education in an Age of AI. Retrieved from teachai.org/cs. [date].

Why is it Still Important to Learn to Program?

Despite the ability of Generative AI (Gen AI) tools to write code, learning to program remains crucial for developing the conceptual understanding, agency, and dispositions necessary to use AI tools effectively, evaluate and modify AI-generated code, and understand the broader context and implications of programming. Programming produces more than code; it encourages creativity, critical thinking, and collaboration and lays a foundation for learning the discipline of computer science (CS). AI might augment learning but cannot replace the understanding and skills gained through learning to program.

Recent advances in Gen AI's ability to write code have led some industry leaders to claim that learning to program is obsolete. Even as professionals use AI tools to generate a growing proportion of code, learning to program continues to be a context for developing the problem-solving and computational thinking skills to use these tools appropriately, evaluate and modify their output, and place the results in context (Salehi et al., 2020). Rather than diminishing the importance of learning to program, Gen AI tools highlight the need for a foundational understanding of programming to comprehend and appreciate these tools.

In the same way that students must learn "number sense" – a conceptual understanding of numbers and how they are related and connected – before they can move on to algebra or calculus, they need to develop "code sense" to be successful in this new era of AI-assisted programming (Johnson, 2024).

⟶ Code Sense

We define **code sense** as the conceptual understanding of a computer program's underlying design, processes, and system relationships, as well as the mental capacity to analyze, simulate, and predict a program's behavior.

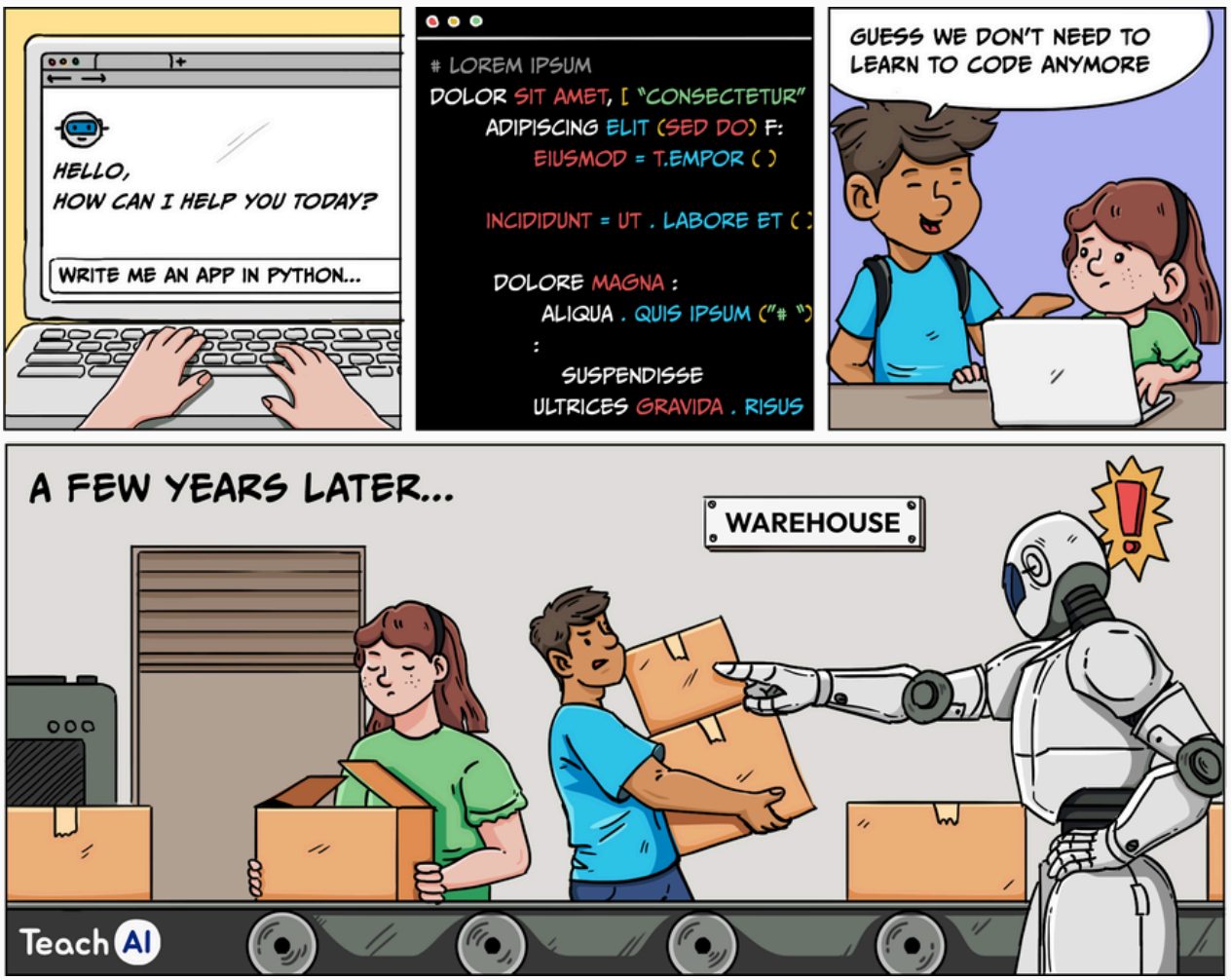
Code sense supports the ability to debug effectively, optimize performance, and understand the broader context of the code within a system.

> "In my vision, the child programs the computer and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building."
>
> - **Seymour Papert**, Mindstorms: Children, Computers, And Powerful Ideas, 1980, p.5

For our purposes, "programming" is used interchangeably with coding and includes aspects of software development. Programming is defined as the creative process of designing, writing, testing, and maintaining code to perform specific functions and solve problems.

In this brief, we dispel the myth that Gen AI coding tools make it unnecessary for students to learn to program.

Learn to Program or Follow Commands: A Comic



AI is not magic. It needs human expertise and guidance.

Myth 1: AI coding tools make programming knowledge and skills unnecessary.

To maximize the value and effectiveness of AI tools, developers need to know how to structure effective prompts, refinements, and requests (Kirova et al., 2024). Learning to program, which is more than learning the syntax of a programming language, gives students opportunities to understand programming logic, practice solving problems, and develop an understanding of algorithms and data structures. This knowledge is necessary to direct AI tools effectively.

For example, early research studying novices aged 10 to 17 years using code generators has shown that those with greater programming fluency benefit more from the tool. This suggests that AI tools are most effective in augmenting, rather than replacing, programming knowledge and skills (Kazemitabaar et al., 2023).

⟶ How Will Programming Evolve in an Age of AI?

From assembly language to modern block-based languages, programming has continuously evolved to be more accessible by abstracting lower-level programming details. Gen AI tools may continue this trend and allow programmers to focus more on creativity, problem-solving, and addressing ethical concerns rather than the mechanics of translating instructions into a programming language. Some programming tasks will be augmented by AI, and some will be replaced.

How should learning to program evolve to reflect the changing nature of programming itself? What insights can CS education gain from how software engineers use Gen AI tools? Sign up for updates to learn more about the future of CS education.

"We need to help students learn to ask questions well. We need to teach them to have ownership of these tools."

- **Christina Gardner-McCune**, Associate Professor, University of Florida and Co-Lead of AI4K12

AI is not perfect. It needs human oversight.

Myth 2: Students do not need to learn to program because AI can do it accurately and independently.

Although Gen AI tools are improving, they are known to suggest code that is inaccurate, contains security vulnerabilities and biases, or is misaligned with a user's intent. A 2023 study of three popular code generators found that they produced correct code between 31% and 65% of the time (Yetiştiren et al., 2023). Another study showed that human and AI pairs critique code more effectively than either can on their own. Learning to program lays the groundwork for developing the code sense and computational thinking skills needed to evaluate, debug, modify, and situate AI-generated code in the context of a larger program.

"AI makes mistakes. Students need to understand the basics first, so they can critically understand what they need from AI and also how to evaluate the results."

- **Neda Blackburn**, STEM Director, Roland Park Country School

⟶ Survey Insight

CS teachers responding to a May 2024 CSTA/TeachAI survey commonly noted that even with AI's prevalence and coding abilities, learning to program is still essential for students to foster the code sense and computational thinking skills necessary to use and build future AI applications.

AI will not replace the need for programmers. It needs human creativity and domain expertise.

Myth 3: Students do not need to learn to program because AI will replace all programming jobs.

The rapid uptake of Gen AI in the workplace is changing programming practices without obviating the need for humans. Learning to program is shifting from focusing on code generation to more code reading, evaluation, debugging, and refactoring (Kirova et al., 2024).

⟶ Focus on the Fundamentals

Gen AI coding tools are rapidly improving to help software engineers by eliminating repetitive tasks, suggesting code, and even translating code from one programming language to another (Stryker, 2024). While industry professionals will continue to adapt to these innovations, students learning to program need to build the capacity to understand the fundamental, durable concepts behind these evolving tools to use them appropriately.

Human creativity and domain expertise have been and will be crucial in programming. Creativity drives innovative problem-solving, intuitive user interface design, and feature development, while domain expertise ensures the solutions are accurate, relevant, and address specific use cases. Together, they enable the creation of robust, user-friendly programs that meet both technical and contextual needs.

> "There is a longstanding need for people who have software development skills. What you need to know may shift – things that were obviously critically important or seen as peripheral will need to change in their relative importance."
>
> - **Ben Shapiro**, Associate Professor, University of Washington

AI can code, but there is more to programming than code.

Myth 4: The only purpose of learning to program is to produce programs; now, AI can do that for us.

Programming goes beyond translating instructions into a programming language; it involves exploring needs, writing specifications, and testing, debugging, and analyzing code. Learning to program offers students a platform for collaborative problem-solving, creative expression, and discovering joy in creating something new (Kafai et al., 2015).

When students work together, they may find it easier not only to absorb new concepts and vocabulary but also to express their curiosity and enthusiasm. Research shows pair programming can boost student retention in CS, particularly among some underrepresented groups

(Campe et al., 2019). Having these opportunities while learning to program nurtures resilient, life-long learners equipped to thrive in a rapidly changing world.

Learning to program also has emotional value. A small-scale study of primary school students showed that children felt significantly happier, more excited, and more in control after learning to program (Tisza et al., 2023). Results like these depend on many factors, especially a high-quality teacher, and they highlight the benefits learning to program can have on students' emotional state, reminding us to consider how AI use may affect students' feelings about CS.

> "Learning how to code isn't just about preparing young people for potential participation in the technology workforce. Learning how to code can be an incredible opportunity for creative expression, enabling learners to share their ideas and voice with the world, bringing joy, satisfaction, and a sense of purpose."
>
> - **Karen Brennan**, Professor, Harvard University

⟶ Computer Science is More than Programming

Learning to program is just one aspect of learning CS. For example, the CSTA K-12 Standards also include understanding how computers work, how messages are sent through the Internet, how data is collected and analyzed, and how technology impacts society. Even if programming were obsolete, a foundational CS experience would still be essential to:

- fueling innovation across multiple fields, such as data science or computational biology,
- addressing cultural and structural barriers to pursuing post-secondary CS,
- expanding access to high-paying, highly satisfying jobs, and
- promoting greater diversity in a myriad of fields (Lewis, 2017).

Three Ways to Take Action

Why is it Still Important to Learn to Program?

Reflection	1. Given that AI can generate code, how can you balance the use of AI tools with the need for students to develop fundamental coding skills and code sense? 2. In what ways can you explicitly emphasize computational thinking skills, whether teaching students programming, writing curriculum, or training teachers? What practical activities or resources can help?
Call to Action	**Administrators** 1. Continue offering introductory CS courses that include programming. 2. Support CS teachers' professional growth through communities such as a CSTA chapter. 3. Support schoolwide professional development and help teachers share lessons learned. 4. Develop guidelines based on resources like the TeachAI AI Guidance for Schools Toolkit.
	Teachers 1. Be explicit about the fundamental concepts and skills you want students to learn (see CSTA K-12 Standards), such as debugging, regardless of whether they use AI tools. 2. Explore using AI tools in the learning process before introducing them to students. 3. Discuss if, when, and how AI tools should be used in CS education. Develop principles to guide AI use in programming assignments.
Further Study	1. How does learning how to program with Gen AI tools compare to traditional instructional methods? How might they help or hurt? 2. How might coding assistant chatbots impact students with less prior CS experience in CS? 3. What are the most common errors and biases found in AI-generated code in educational settings? How can educators train students to identify, evaluate, and correct these issues?

TeachAI is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum.

How Are Computer Science Educators Teaching With and About AI?

Computer science (CS) is a natural context for students to explore AI's benefits, limitations, and societal impacts, and CS teachers overwhelmingly want professional development to facilitate these learning opportunities effectively. Many teachers already teach with and about AI in a CS context to emphasize computational thinking, demystify AI, and equip students to use AI tools responsibly.

As AI becomes pervasive in students' lives in everything from recommendation systems to voice assistants, CS teachers are incorporating lessons to demystify AI and encourage a critical mindset toward its benefits, limitations, and societal impacts. CS teachers are using AI tools to help explain code, provide timely feedback to students, and help them break through "coder's block" and syntax challenges so that they can focus on the creative aspects of programming. All teachers should follow existing policies, such as age restrictions based on privacy or security concerns, and consider the appropriateness of AI tools for specific activities even when the tools are permitted. There is a growing body of unplugged resources to help students learn about AI when policies or limited technological infrastructure prevent AI use.

In this brief, we examine how teachers teach with AI, such as using AI as a tool for student learning or teacher support, and teach about AI, such as how it works and its ethical and societal implications.

Survey Highlights

Guidance on Computer Science Education in an Age of AI is informed by a survey of CS teachers (n = 364 teachers, 24% primary, 76% secondary, 12% international) administered by the CSTA and TeachAI in May 2024.

of teachers think students in introductory courses should learn about AI.

of teachers said they feel equipped to teach about AI.

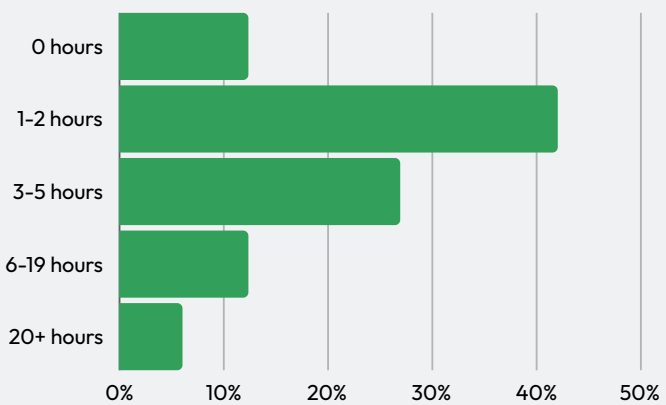
of teachers said they would benefit from professional development to learn how to use and teach about AI.

Teaching About AI

CS teachers are incorporating AI into discussions, demonstrations, and unplugged activities to demystify AI and make it more approachable.

CS teachers are teaching students how AI works and discussing AI's ethical implications. Most teachers (79%) agree that curriculum and standards should be updated to emphasize AI.

⟶ How much do students currently learn about AI in your computer science curriculum?



Teachers indicated that foundational CS concepts like algorithms help students develop the problem-solving skills and awareness of societal impacts required to use AI tools critically and to create with them responsibly. Classroom activities about AI can examine topics such as:

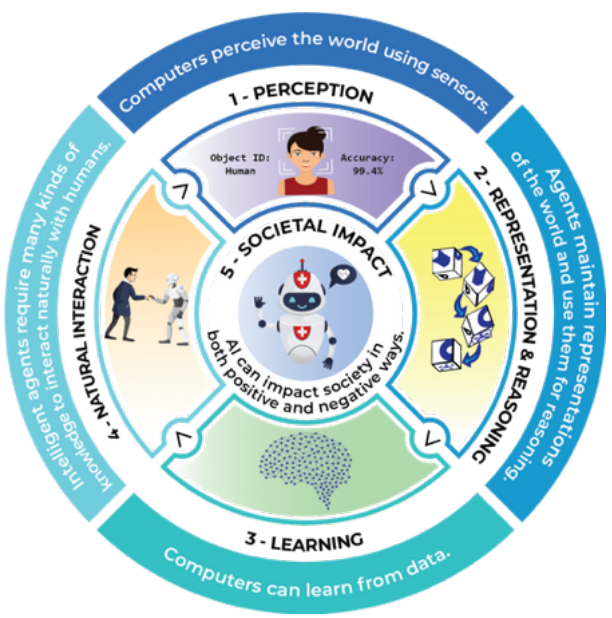
- underlying programming and algorithmic concepts,
- common uses,
- limitations, and
- ethical use and impact on society (Lee & Kwon, 2024).

What's more, topics such as algorithms, classifiers, natural language processing, and AI's ethical issues extend beyond the CS classroom and are relevant in mathematics, language, biology, and social science (Casal-Otero et al., 2023).

> "I do hope that educators realize the importance of facing the reality of AI as well as understanding that the goal of a paradigm shift like this should not be about policing students but about enabling students to think about using [and] understanding this new technology."
>
> - **Ben Smith**, CS Teacher, Rowland Hall School

⟶ AI4K12

Since 2018, the AI4K12 Initiative has been developing national guidelines for teaching about AI, organized around the 5 Big Ideas in AI.



AI in Primary CS Education

Vicky Sedgwick, a primary school educator, integrates machine learning into lessons that address the CSTA K-12 Standards on data and analysis. Students sort, classify, and label data and use the data to train a simple model. Students can then predict how the model

might classify new pieces of data and explore how machines learn and begin to recognize bias in data. AI concepts like natural language processing and image recognition also fit into algorithms and programming standards. Elementary and middle school students can explore how AI works with resources such as Code.org's AI for Oceans, Machine Learning For Kids, or MIT RAISE Playground.

There is also a growing body of unplugged activities that do not require access to AI or a computer and are available for teaching AI concepts. For example, AI technology playing cards teach about input, algorithm, and output to demonstrate how developers' choices affect results (Long et al., 2021).

Revised CSTA K-12 Standards, slated for release in summer 2026, will include AI-related learning goals. In the meantime, the Reimagining CS Pathways project has defined foundational CS content that includes learning progressions specific to AI.

> "[T]here is a tendency to make AI seem either magical, sentient, infallible, or overly human . . . Since such (mis)representations are rife in mainstream discourse, K-12 education needs to work extra hard to address this challenge through approaches to demystify AI and lift the hood on how it works ." (Grover, 2024)
>
> - **Shuchi Grover**, Director, Looking Glass Ventures and Edfinity

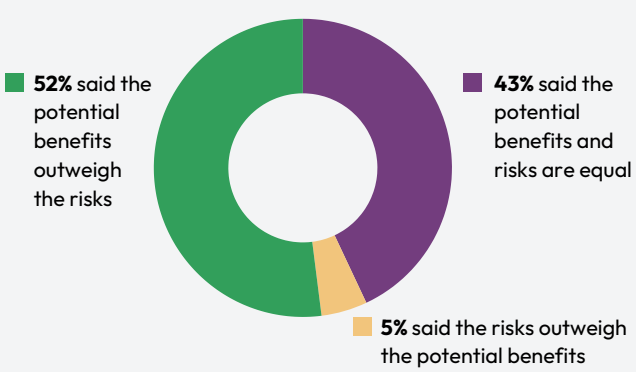
Teaching with AI

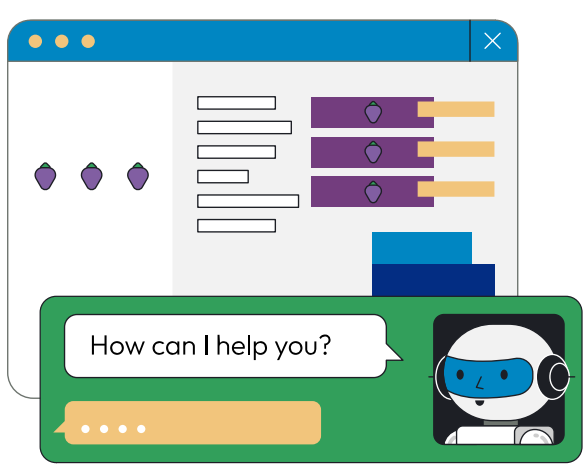
CS teachers are integrating AI tools into instruction, assignments, and assessment.

A systematic review of K-12 AI education from 2018 to 2023 describes a variety of early initiatives to teach with AI by using machine learning tools and AI platforms to help students understand basic concepts, such as training data, testing models, and identifying bias (Lee & Kwon, 2024). Hands-on experiences with AI tools not only helped students develop an understanding of AI concepts but also made learning more fun. Although the specific way that AI is used in the classroom is an essential factor, the review also suggested that using AI in the classroom can help students:

- develop foundational CS skills,
- reflect on critical and ethical considerations about the role of AI and its use,
- apply AI knowledge to real-life problems, and
- improve motivation and interest in technology.

⟶ Thinking about ways AI tools could be used in CS education...





AI Feedback

Maria Camarena, a secondary CS teacher in California, uses ChatGPT to improve students' programming skills. Students input lines of code they have written and a description of the code and receive programming and writing feedback. This approach strengthens programming skills and supports multilingual students with their English language development.

Early research also identifies opportunities to use Large Language Models (LLMs) to improve the CS learning experience and foster critical-thinking skills, including:

- generating code as a starting point for a solution,
- teaching students how to describe solutions in natural language, and
- guiding problem-solving strategies (Prather et al., 2023).

LLMs have also been shown to help provide a vast selection of solutions of varying quality so students can practice critical analysis by examining multiple answers to a problem (Denny et al., 2023).

In one early effort to revise a post-secondary introductory CS curriculum to use LLMs, problem decomposition and explaining/testing/debugging code were emphasized over syntax and writing code (Vadaparty et al., 2024). The scope of students' projects transcended what was typically seen in a CS1 course, and their exam results reflected results similar to previous CS1 classes.

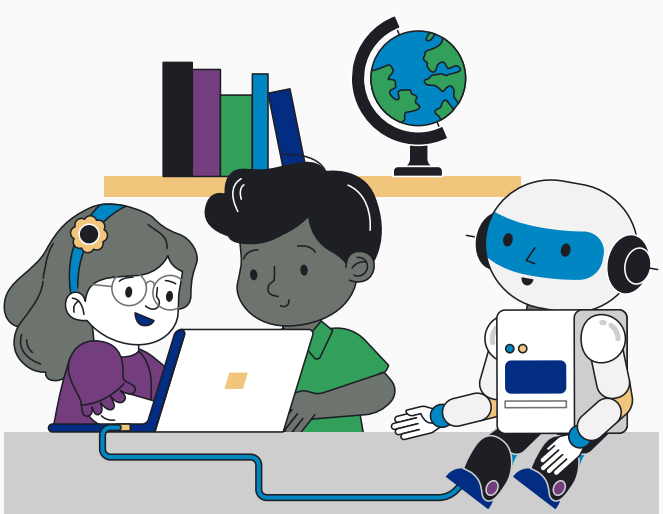
Pairing with AI

Michael Phelan, a CS teacher in North Carolina, allows students in his Python 2 class to use ChatGPT or Copilot to give them an experience similar to pair programming while pushing them to improve their own thinking. However, it is vital to Phelan that students have a comprehensive understanding of what they are doing rather than relying on AI to do the work.

Learning from Math Education

Lori Jacques, a former math teacher and current CS professor, describes three approaches based on math pedagogy to help students look at problems differently and improve their programming skills and conceptual understanding (Jacques, 2023):

- **Create Multiple Representations**: Students can use Generative AI (Gen AI) to generate code, explain code in plain language, illustrate how the output changes as the code executes using a flow chart, and compare the different representations and their pros and cons.
- **Explore Different Approaches**: Students can use AI code generators to create different programs to address the same problem and then compare them to each other and human-generated solutions, including their own.
- **Explain Code Created by AI**: Students can analyze, describe, and explain the functionality of AI-generated "worked examples," which may or may not be correct.



Concerns About AI's Risks

CS teachers responding to the CSTA/TeachAI survey frequently cited these three concerns about the potential risks of using AI in the classroom:

- AI will do the thinking for students.
- Students will become overreliant on AI tools.
- Students will lose essential skills.

These concerns can be mitigated by teaching foundational programming skills and critical engagement with AI tools. This approach ensures that students use AI as a supportive tool, maintaining their problem-solving abilities and understanding the underlying logic.

"Students have a tendency to 'trust' the AI and blindly copy and paste the code but lack the skills to 'talk to' the code with AI and truly understand what is going on. This causes them frustration when later attempts don't work, and they lack the vocabulary or experience needed to properly explain what problems or tasks need to be done to their AI assistant."

- **Graham Nolan**, CS Teacher, Hong Kong International School

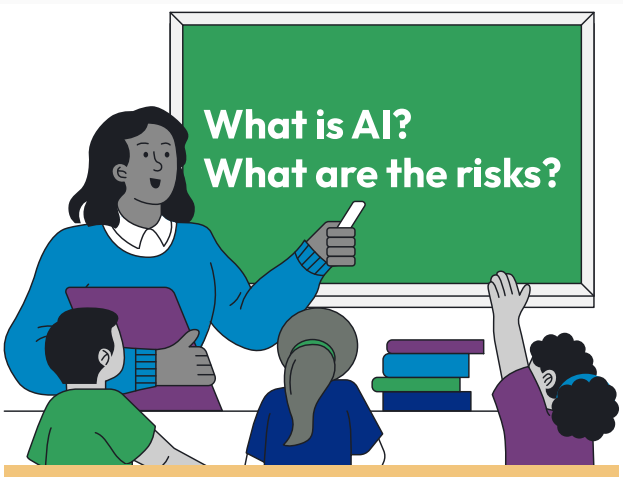
For more on the ongoing need for students to learn CS fundamentals, see Why is it Still Important to Learn to Program?

Teacher Support

CS teachers are using AI to support and streamline their own work.

Teachers who participated in the CSTA/TeachAI survey reported using AI to facilitate their work in various ways, including tailoring and refining lesson plans; providing differentiation for students who need either more challenge or more support; and organizing ideas, breaking them down, and re-explaining them.

These uses mirror common opportunities for Gen AI to streamline teacher workloads. In one early study, researchers prompted the code generator Codex with source code to generate new programming exercises. They found that a large majority were completely novel and aligned with the topic and theme at hand (Denny et al., 2023). In addition to saving instructors' time, the variety of solutions that code generators can create from one input can offer students a view into different approaches to tackling the same problem.



"While AI can automate certain tasks, such as grading or feedback provision, overreliance on AI tools may diminish educators' role in the teaching process. It's essential to strike a balance between leveraging AI for efficiency and maintaining the human element of teaching, including personalized interaction and mentorship."

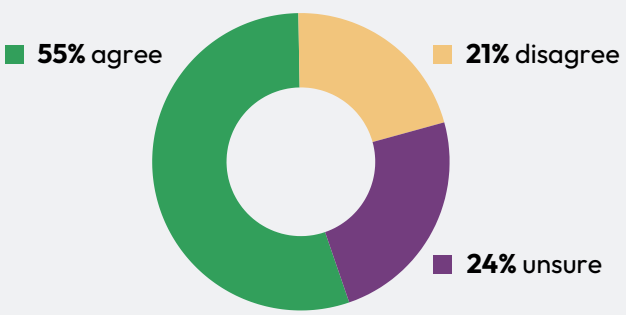
- **Fred Sagwe**, CS Teacher, Shimo La Tewa School and Robotics Society of Kenya

Inaccuracy and Overreliance

Despite the potential benefits of using AI tools to enhance learning and provide instructor support, common concerns involve Gen AI's tendency to produce inaccurate information and students' ability to make sense of the output. Both students and teachers also express concerns that students may not learn their material when they turn to Gen AI for help (Zastudil et al., 2023).

Teachers have mixed opinions about whether students in introductory CS classes should learn to program with AI. The benefits and risks of learning to program with AI tools are still being discovered and will depend on the classroom context and how the tools are used.

⟶ Students in introductory CS classes should learn to program with AI.



⟶ Global Perspectives

As of July 2024, 41 state education agencies in the U.S. and over 25 government agencies globally participate in TeachAI. Here are some global perspectives on the future of CS education:

"When we teach students how to understand and responsibly leverage AI, we equip the next generation of technology users and creators with skills and confidence to innovate."

- **Oksana Pasichnyk**, CS Teacher, Lyceum Sykhivsky, Ukraine

"AI in coding classrooms faces challenges in implementation due to varying teacher and student understanding and skill levels. Effective AI integration requires teacher and student training, well-designed learning experiences, ethical guidelines, and ongoing support. Despite these challenges, AI offers immense potential to transform coding education in Thailand."

- **Cheeraporn Sangkawetai**, Institute for the Promotion of Teaching Science and Technology, Thailand

Three Ways to Take Action

How Are Computer Science Educators Teaching With and About AI?

Reflection	1. In what situations might teaching with and about AI be useful? In what situations might it be inappropriate? 2. Where can teaching about AI be integrated into the existing curriculum? 3. What changes can be anticipated in students' engagement, motivation, or understanding of programming concepts after introducing AI tools?
Call to Action	**Teachers** 1. Participate in ongoing professional development (PD) to deepen knowledge and skills in teaching with and about AI. Find PD from CSTA, CSTA chapters, and curated programs. 2. Follow local guidance on the use of AI tools in classrooms. When local policies do not exist, develop guidelines for using AI tools on classroom assignments. Prioritize responsible practices that encourage students to determine if, how, when, and why they would use AI. 3. When using AI in your work, constantly evaluate the output to see how it fits your classroom context, objectives, and teaching style.
	Curriculum and Professional Development Providers 1. Include AI as a topic and tool and explore uses of AI that will enhance activities, examples, lessons, and projects. Proceed cautiously, with an eye towards safe and responsible use. 2. Provide professional development enabling teachers to experiment, share use cases and insights, and field questions and concerns in their schools. 3. Ground questions and projects for students in real-world examples that also illustrate the benefits and risks of AI.
Further Study	1. How should CS curriculum and standards be updated to include AI concepts and tools? 2. How can school systems address barriers to access, support effective pedagogical design, and provide powerful learning opportunities to provide equitable access to AI education? See **Classroom Perspectives on AI** (TeachAI, 2024) for more information on these issues. 3. How do the interactions between teachers, students, and AI tools influence the teaching and learning process and outcomes?

TeachAI is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum.


MINECRAFT
DELL

How Can Students Become Critical Consumers and Responsible Creators of AI?

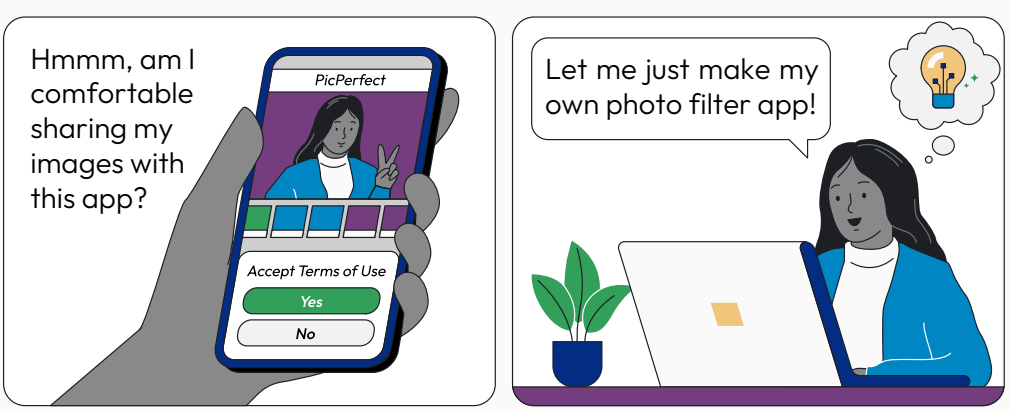
Teaching students to become critical consumers and responsible creators of AI involves integrating ethical and societal considerations into computer science (CS) education. CS teachers are incorporating discussions about transparency, accountability, information accuracy, privacy, fairness, and ethical design into their curricula. These lessons empower students to evaluate AI outputs critically, understand the implications of AI on society, and design technologies that prioritize equity and responsibility. More than any other subject, CS provides an opportunity to explore AI's societal and ethical impacts by learning how AI models are trained, understanding how they work, and developing their own models.

From producing deepfakes to amplifying gender, racial, and cultural biases, AI's potential to harm individuals and communities is in the spotlight, creating increasing demand for incorporating societal and ethical considerations into CS education.

An essential component of teaching students about AI is empowering them to understand how AI models are built and to think critically about the benefits, harms, possible safeguards, and the people and communities affected by this powerful technology. As more CS education involves consuming and creating AI, students must also learn to ask whether and why they should use AI in the first place.

> "When it comes to AI education, we do not have the luxury of burying our heads in the sand. CS teachers have the opportunity and responsibility to lead students in understanding the societal and ethical implications of AI: the good and the bad, the benefits and harms, the possibilities and realities."
>
> - **Charity Freeman**, CSTA Board Chair

Critical Consumers, Responsible Creators

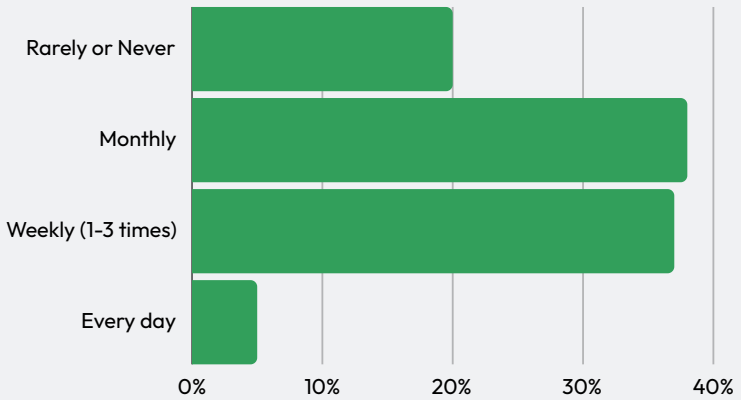


According to Reimagining CS Pathways: High School and Beyond, the societal impacts of computing and related ethical implications are the highest-rated topics of importance across the CS education community.

Additionally, 66% of CS teachers responding to a CSTA/TeachAI Survey (May 2024) said that conversations about AI have helped students understand the societal and ethical issues surrounding AI. Ethical concerns about AI also emerged as one of the most requested professional learning topics.

In this brief, we highlight a few of AI's ethical and societal considerations and offer some practices for developing critical AI consumers and responsible AI creators.

⟶ How often do you supplement your CS curriculum to include discussions of ethical questions related to AI, such as algorithmic bias, digital citizenship, and privacy?



Transparency and Accountability

AI systems are widely perceived as opaque, containing unknowable data sources and inscrutable decision-making processes.

When equipped with explanations about how AI systems work, teachers are less reluctant to use them in their classrooms and less concerned about the technology's impact on personal autonomy. Early research suggests that teachers are more likely to trust AI when given the ability to review and override AI recommendations (Nazaretsky et al., 2022).

Classroom Practice: Create model cards that explain the intended use, training dataset, accuracy, and limitations of a classification or prediction model.

When asked about the potential risks of including AI in the classroom, CS teachers responding to the CSTA/TeachAI Survey emphasized that students must be made aware that AI makes mistakes and can have disparate societal impacts, often perpetuating harm on marginalized communities. These same concerns were commonly cited as reasons for students to deepen their core CS knowledge.

> "Students are more empowered to engage in popular AI discussions when they understand the technical side of the ethical and societal impacts of AI."
>
> - **Sepehr Vakil**, Associate Professor, Northwestern University

Information Accuracy and Integrity

AI makes it easier for unethical parties to produce misinformation and disinformation by mimicking voices, images, and videos.

During the 2023-24 academic year, parents and students in California, New Jersey, and Washington sounded the alarm over AI: Classmates used AI-based image generators to create explicit "deepfakes" from pictures online and circulated the images on social media. Districts responded by updating their social media guidelines and educating students about the appropriate and acceptable use of Generative AI (Gen AI) tools while urging others to learn more about misinformation. There is a broad range of content authenticity issues, from creative works that imitate specific artists to deepfakes in the democratic process, which raise significant ethical and legal concerns about originality, consent, and the potential for misinformation.

Classroom Practice: Discuss the importance of verifying information sources and explore ways to validate content. Explore instances of when AI output adds value to individuals and society and when it does not. Discuss the importance of citing sources when using AI.

"Students can potentially misuse AI for themselves and others, but teaching them and being open about the benefits and harmful effects and potential upsides and downsides far outweigh the potential risks."

- **Art Lopez**, Project Manager, Coding Our Future, University of California, San Diego

"The hard problems of AI and ML education come down to civic values, philosophy, ethics, fairness, bias, transparency, explainability. The social studies teachers - we can learn from them."

- **Ben Shapiro**, Associate Professor, University of Washington

Privacy Rights

Data collected by AI-powered systems and educational tools raise concerns about students' privacy rights.

In March 2024, a coalition of 41 educational and civil rights organizations wrote to the U.S. Department of Education, asking to end the funding of AI-powered security systems in K-12 schools. They were concerned about AI's potential to violate students' civil rights. Especially alarming, the group said, was the increasing use of facial recognition technologies, social media surveillance, behavioral threat assessments, predictive policing, and other surveillance tools.

As students turn over growing volumes of personal data, including data collected by educational tools, they must understand the potential for combining it to divulge sensitive information. Moreover, tracking and monitoring components in AI systems that predict student performance, weaknesses, and strengths can inhibit their participation at school (Akgun & Greenhow, 2021).

Classroom Practice: Discuss how, when, and why students' personal data are collected, analyzed, interpreted, and used. Consider how decision-makers might use this data to their advantage — even if it is something students are not okay with. Discuss students' rights and how they can opt out of data collection.

"Anyone developing technologies has a responsibility to keep ethics top of mind and create things that will not cause harm. There should be a pledge or agreement that these persons go through so that 'doing no harm' is at the basis of their profession. . . This must also be followed with action."

- **DaQuan Bashir**, Professional Learning Manager, CSTA

Fairness and Justice

AI systems' disparate impacts on different demographic groups raise alarms about fairness and justice in our digitally driven society.

AI systems can reflect and amplify the implicit biases of their creators. Understanding AI biases involves understanding how people create technologies like Large Language Models (LLMs) and how biases can creep into all stages of development, such as selecting training data and selecting which features in the training data to prioritize (Broll & Grover, 2023). Examples of discrimination against women, people who are economically disadvantaged, or people of color as a result of algorithmic bias have shown up in hiring, health care, insurance, and mortgage lending decisions, in addition to criminal justice assessments.

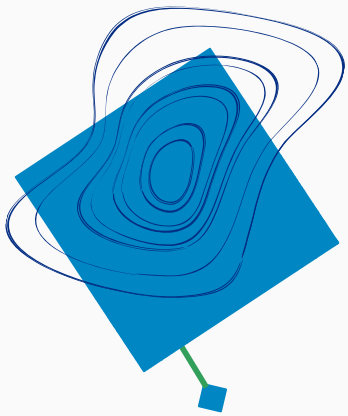
The non-profit journalism group ProPublica in 2016 was the first to uncover racial bias in COMPAS (Correctional Offender Management Profiling for Alternative Sanctions), an AI-powered system used to predict a criminal defendant's recidivism risk. COMPAS falsely identified Black defendants as likely future criminals nearly twice as often as white defendants. The AI model was trained on historical crime data, and, as a result, individuals who were part of a demographic historically targeted by law enforcement were likely to receive higher recidivism scores. A related issue raised by the COMPAS example is transparency, as the designers of COMPAS refused to reveal details of the software and its algorithms.

Classroom Practice: Discuss the pros and cons of using AI for different projects and consider when another solution might be a better option. Discuss how AI tools might be used to detect and measure bias in datasets.

"[A]ll young people need to understand whether the AI with which they knowingly or unknowingly engage has treated them fairly." (Holmes et al., 2022, p.31)

⟶ Responsible AI & Tech Justice

The Kapor Foundation's Responsible AI & Tech Justice: A Guide for K-12 Education helps educators develop learning experiences that prioritize critically evaluating technologies and creating more equitable solutions. One recommendation is to "examine the AI technology creation ecosystem from who designs and develops products and how they are developed, to who invests in their creation and benefits from their adoption."



Ethical Design

Transparency, accountability, equity, and privacy should be principles in technology development.

Ethical design recognizes that all technologies are embedded in social systems and reflect designers' beliefs. Ethical design considers societal values, includes potential users throughout the design process, especially those most affected by the design, and follows principles of transparency, explainability, accountability, equity, and privacy.

In a study of recommendation algorithms and ethical design, students as young as 5th graders were shown to be capable of recognizing design agendas and identifying influences on those agendas (DiPaola et al., 2020). Students were asked to identify relevant parties associated with YouTube and their potential values, encouraging questions about the data collected from the platform, how it influences what they watch, and how it affects their beliefs. Students then had an opportunity to develop their own design goals and agendas using ethical design principles.

Classroom Practice: Early in the design process, have students explore different user personas: representative examples of potential users with different backgrounds, desires, ability levels, and perspectives. For more information, see Practice 1: Fostering an Inclusive Computing Culture in the K-12 CS Framework.

⟶ AI4K12: Societal Impact

AI is a disruptive technology that will not only impact the economy and employment but also shape social and cultural norms (AI4K12 Framework's Big Idea 5).

Ethical Concerns and Practices

Topic	Ethical Concern	Classroom Practice
Transparency and Accountability	AI systems are widely perceived as opaque, containing unknowable data sources and inscrutable decision-making processes.	Create model cards that explain the intended use, training dataset, accuracy, and limitations of a classification or prediction model.
Information Accuracy and Integrity	AI makes it easier for unethical parties to produce misinformation and disinformation by mimicking voices, images, and videos.	Discuss the importance of verifying information sources and explore ways to validate content.
Privacy Rights	Data collected by AI-powered systems and educational tools highlight student privacy issues.	Discuss how, when, and why students' personal data is collected, analyzed, interpreted, and used.
Fairness and Justice	AI systems' disparate impacts on different demographic groups raise concerns about fairness and justice.	List the pros and cons of using AI for different projects and consider when other solutions might be a better option.
Ethical Design	Transparency, accountability, fairness, and privacy are crucial in technology development.	Address the needs of diverse users and consider when the risks of using AI might outweigh the benefits for an individual or community.

Three Ways to Take Action

How Can Students Become Critical Consumers and Responsible Creators of AI?

Reflection	1. How can you incorporate ethical and societal issues related to AI into CS content? How can you address the potential challenges when engaging in sensitive topics? What can be learned from other subjects? 2. How can you model the ethical and responsible use of AI tools? What steps can students take to understand AI tools' limitations and biases and evaluate their outputs critically? 3. What data do you and students share on online platforms and with AI systems, both knowingly and unknowingly? What risks are associated with sharing personal data?
Call to Action	**Standards Writers** 1. Ensure that ethical considerations and societal impacts of AI and technology are embedded in primary and secondary education standards. 2. Connect the societal and ethical impacts of AI to how AI works and how people develop AI technologies. 3. Develop standards that integrate CS education with other disciplines, such as science, social studies, philosophy, and ethics, to provide a holistic view of AI's societal impacts.
	Curriculum and Professional Development Providers 1. Include deliberate conversations about bias, ethics, and AI's impact throughout the curriculum. Provide students with reflection questions at the start and end of each project. 2. Encourage perspective-taking by asking about user stories and how tools might work. Frame the potential for positive, negative, and unintended consequences. 3. Create space for honest conversations and thoughtful questions. Include questioning protocols to foster deeper thinking about how what people design will affect others.
Further Study	1. How does incorporating ethical considerations, such as bias and fairness, into CS content affect students' understanding and critical analysis of AI technologies? 2. What are effective methods for teaching transparency and accountability? How do these methods influence students' trust and willingness to engage with AI technologies? 3. How can educational interventions be designed to effectively reduce biases and promote fairness in students' AI projects and understanding?

TeachAI is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum.

The Role of AI in Computer Science Education
⟶ Results from a Teacher Survey

Key Takeaways

1 **CS teachers want to teach with and about AI.**

Most teachers (85%) believe using and learning about AI should be included in foundational CS experiences, and (80%) agree that CS standards should include AI. Additionally, 87% say students should learn about careers impacted by AI.

2 **CS teachers want support in teaching with and about AI.**

A notable 88% of teachers expressed the need for professional learning and additional resources to effectively teach with and about AI, highlighting the necessity for updated professional development programs.

3 **CS teachers are mostly optimistic or neutral about the potential benefits and drawbacks of AI in CS education.**

While 52% believe the benefits of AI outweigh the risks, 43% see an equal mix, and only 5% said the risks outweigh the benefits. Key concerns include overreliance on AI, the potential for plagiarism, and ethical issues such as bias in AI systems.

4 **CS teachers stress the importance of teaching programming skills alongside AI education but disagree on when students should start using AI tools.**

Teachers believe programming is essential for understanding, debugging, and effectively collaborating with AI and crucial for developing computational thinking, problem-solving, and critical thinking skills.

While 55% of CS teachers believe students in introductory CS classes should learn to program with AI, 24% are unsure, and 21% disagree.

5 **CS teachers cite bias, misinformation, overreliance, and the ethical implications of Gen AI technologies as common concerns.**

Across the survey, teachers identified the need to talk with students about AI's ethical and societal impacts. They expressed the desire for guidance on discussing these issues and how AI tools will specifically impact the classroom.

CSTA and TeachAI conducted a survey in May 2024 about computer science teachers' usage of, beliefs about, and attitudes toward AI.

- 364 responses from current CS teachers
- 24% primary school, 76% secondary school
- 12% outside of the US
- 44% urban, 43% suburban, 13% rural

CSTA will gather additional data on these questions in a teacher landscape survey in the fall of 2024.

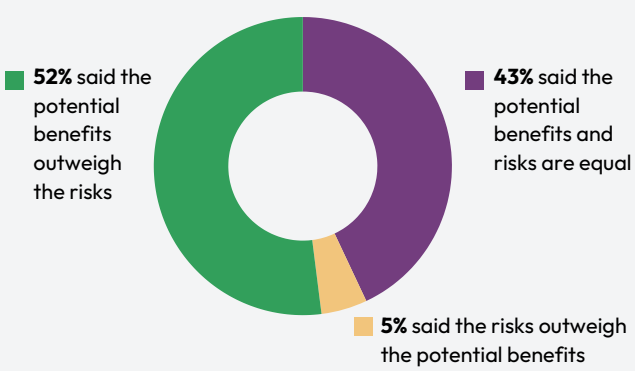
AI Curriculum Content

Why teach with and about AI?

Teachers emphasized that understanding AI is essential for students' future personal and professional lives in the digital age. 76% agree that AI can be used to promote creativity in the CS classroom.

Teachers were mostly optimistic or balanced about the benefits and drawbacks of using AI in the classroom.

⟶ Thinking about ways AI tools could be used in CS education...



What should we teach about AI?

There is a strong consensus among CS teachers about the importance of AI education:

- 87% agree that students should learn about careers impacted by AI.
- 79% believe that CS standards should be updated to include discussions about AI.
- 85% think students in introductory courses should learn about AI.

How should CS curriculum evolve?

Teachers indicated that students should continue learning fundamental programming skills, including topics like variables, loops, conditionals and topics that directly apply to AI, like data structures and algorithms. Many strongly advocated for students to learn about the societal impacts of AI in their foundational CS courses. Many also responded that they were unsure which CS topics should be emphasized or changed.

> "There is no getting around the need for humans to master order of execution, variables, logic, conditionals, looping, functions, modular design, and so on."
>
> - **Secondary CS Teacher**, Maine, USA

"I don't think any topics should be removed. But I think that real world applications and the future need to be emphasized more... I believe that AI needs to be included into the standards as soon as possible."

- **Secondary CS Teacher**, Pennsylvania, USA

"Collaboration should always be emphasized. Struggling with a problem to find a solution should be emphasized. Understanding that there are multiple ways to solve a problem should be emphasized."

- **Secondary CS Teacher**, Maryland, USA

"I think topics should include more about the mechanics of computer science, the changing job landscape, daily life intersection with computer science, and AI fundamentals for students."

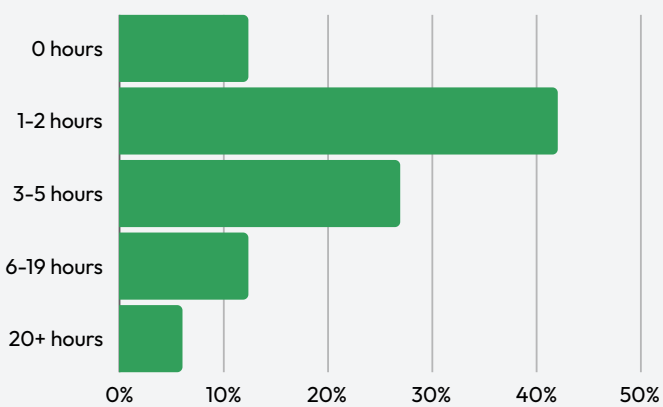
- **Primary and Secondary CS Teacher**, Ohio, USA

Teaching Methods and Support

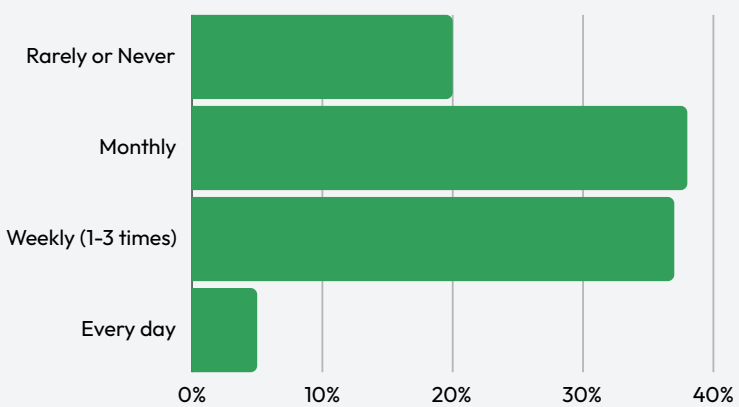
How often are teachers teaching about AI?

CS teachers are teaching students how AI works and discussing AI's ethical implications with students.

→ How much do students currently learn about AI in your computer science curriculum?



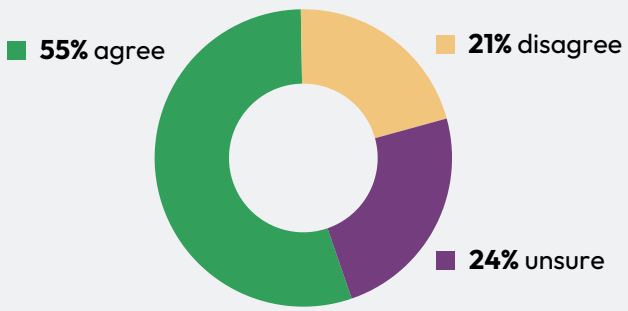
→ How often do you discuss ethical questions related to AI, such as algorithmic bias, digital citizenship, and privacy?



When should AI be used in the classroom?

There is a division among teachers on when to introduce AI tools.

⟶ Students in introductory CS classes should learn to program with AI.



Interestingly, 54% of secondary school teachers agreed that students should learn to use AI in introductory programming courses, versus 62% of primary school teachers.

What AI topics should be included in CS teacher professional development?

Teachers want to integrate balanced discussions about the benefits and risks of AI with hands-on projects, but they require additional support to teach with and about AI effectively.

- 48% feel currently equipped to teach AI.
- 88% need more resources for professional learning.
- 4% disagree that they need additional support.

The most commonly requested topics for professional development centered on how AI works, how to use AI, and the ethical impacts of AI tools. While these topics are not exclusive to the CS classroom, teachers identified their relevance to CS learning and their importance to developing digitally literate students.

"I want to be able to spend my time gathering new resources to use in my classroom and less time grading papers and creating worksheets and exams."

- **Primary CS Teacher**, New Jersey, USA

"For professional development tailored to computer science teachers, focusing on AI topics like machine learning fundamentals, ethical considerations in AI development, integrating AI concepts into curriculum, and practical coding applications would be crucial."

- **Secondary CS Teacher**, Bauchi, Nigeria

"I would love to see more PD for elementary level CS teachers with ways to ready these young learners without necessarily using Gen AI. Also, teachers need PD more readily available for themselves. Our district thus far has provided none, so unless a teacher is highly motivated to seek understanding themselves, they are not getting it."

- **Primary and Secondary CS Teacher**, Arizona, USA

Student Learning and Development

Why should students still learn to program?

Teachers most commonly responded that given Gen AI coding tools, students should still learn to program to be able to develop new Gen AI applications, understand how AI works, and ground their work in fundamental computer science knowledge.

> "Learning computer science is still crucial for students, even with the rise of AI. Understanding the principles behind coding and computational thinking not only enables students to create AI but also helps them grasp its limitations, ethical considerations, and potential societal impacts."
>
> - **Secondary CS Teacher**, New York, USA

Teachers emphasized that developing a robust "code sense" through learning to program is just as crucial with AI as without. Many teachers noted that AI does not generate perfect code, and students must have a solid foundation in programming to analyze, debug, and use what AI creates.

> "As with many parts of education, students need to learn the mindset and logic of programming. This allows them to fix mistakes or optimize poorly written code, whether that is from a classmate or an AI."
>
> - **Secondary CS Teacher,** Kentucky, USA

Teachers also championed the "programmer's mindset" and shared about the value of computational thinking, problem-solving, and creative skills developed in the CS classroom.

> "AI is no substitute for human ingenuity. Students still need to learn problem-solving, logic, teamwork, communication, etc., which is the heart of computer science."
>
> - **Secondary CS Teacher**, Massachusetts, USA

> "Coding teaches students to think computationally, problem-solve, think through steps, cause and effect, debugging (finding and fixing their mistakes), and critical thinking skills. These skills are not caught; they are taught and learned."
>
> - **Primary CS Teacher**, New Jersey, USA

What are the potential benefits of using AI in the CS classroom?

CS teachers most commonly identified the potential benefits of using AI in their classrooms as increasing learning opportunities, automating tasks, and preparing students for their futures in the digital world.

- They most commonly identified AI as helpful in enhancing students' learning opportunities by strengthening engagement, broadening access, and shaping their content knowledge.
- They described AI's ability to differentiate instructional materials, including translation support for multilingual learners.
- They also noted that AI's ability to provide personalized feedback and additional examples leads to a deeper understanding of information.

> "It can be used to model or explain code and concepts to scaffold a student's learning. It can create additional examples for students to explain, create faulty code for 'find the errors,' recommend sources for research, create sample questions on a topic to help test-prep."
>
> - **Secondary CS Teacher**, New York, USA

CS teachers praised AI's ability to automate tasks, such as designing assessments and lessons and providing feedback on student work. With support from AI in their learning management systems and everyday classroom procedures, teachers can focus more on the students in front of them.

> "AI can free up your time for more one-on-one instruction by grading repetitive tasks and providing instant feedback on student code."
>
> - **Secondary CS Teacher**, California, USA

By introducing students to AI early — how it works and how to use it — teachers prepare students for their future personal and professional lives in the digital age.

> "[We are] preparing students for the future where AI is a tool they can use and work with, not against or replaced by."
>
> - **Secondary CS Teacher**, Maine, USA

Ethical and Practical Considerations

What are the potential risks of using AI in the CS classroom?

The most common concerns were about overreliance on AI, plagiarism, and biases or ethical concerns. CS teachers were particularly concerned that students would use AI instead of learning the content, losing both programming skills and the collaborative experience.

> "Students will not have to 'struggle' with problems. They can use AI to help debug code or outline a program. Students will use AI for that collaborative problem-solving rather than engaging with peers or instructors. Communication skills are really important for students to develop."
>
> - **Secondary CS Teacher**, Montana, USA

CS teachers also described that, without such understanding, students might be more inclined to use AI to cheat on their assignments.

> "Without supervision or intentional usage, it could be misused to create code that students claim as their own."
>
> - **Secondary CS Teacher**, Kentucky, USA

Teachers also expressed concerns that biases in AI algorithms might impact the classroom community.

> "AI systems have proved to be biased and to produce incorrect coding solutions at times. It could lead to incorrect or biased learning for the students, as well as discriminated grading if an AI tool is used for grading."
>
> - **Secondary CS Teacher**, Arizona, USA

Finally, teachers discussed the importance of student privacy and protecting sensitive information. While they agreed that AI might benefit students by providing personalized support, they emphasized the importance of caution and clearly understanding what data is being collected or used and how.

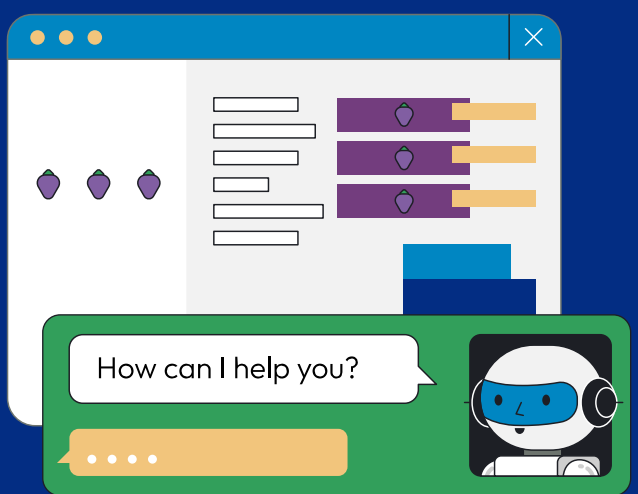
TeachAI is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum.

About TeachAI

TeachAI brings together education leaders and technology experts to assist governments and education authorities in transforming education through teaching with and about AI. The initiative is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum, and advised by a diverse group of 150+ organizations and governments. TeachAI's goals include guiding policy, increasing awareness, and building community and capacity.

About CSTA

The mission of the Computer Science Teachers Association (CSTA) is to empower, engage, and advocate for K-12 computer science teachers worldwide. With more than 25,000 members, CSTA supports and promotes the teaching of computer science and other computing disciplines.



Visit teachai.org/cs

TeachAI is led by Code.org, ETS, the International Society for Technology in Education, Khan Academy, and the World Economic Forum.

TeachAI & CSTA (2024). Guidance on the Future of Computer Science Education in an Age of AI. Retrieved from teachai.org/cs.