Why Study Computer Science In The Age of AI

Stefan Gerber
California State Channel Islands

Abstract

Why would you study Computer Science when AI is so prevalent? This short paper attempts to address this question by collecting somewhat empirical evidence, arguing towards the defense of the value of a modern Computer Science education. Analyzing general trends in the current workforce, discussing the productivity gains of AI, and evaluating the learned computational mindset, allows for a reasonable case to be made in the defense of continuing to learn Computer Science. The conclusions drawn from this paper are not meant to serve as actionable advice, but rather as a unique perspective on the situation.

1 Introduction

When looking at the extreme advancements conducted within the artificial intelligence space, it might seem that the value of a degree, especially within Computer Science, has been significantly devalued. Within a world in which most tasks are indeed handled through automation and artificial intelligence, this seems to be a logical conclusion. However this position could also be construed as being naive or too focused on the sensationalism of the topic.

2 The Purpose of Computer Science

For many within the general population, Computer Science appears to be mostly concerned with becoming a programmer, coder, or developer. This is not necessarily true and represents a discrepancy within the zeitgeist and the actual nature of the profession, confusing what is essentially a basic skill for the entirety of the field. Within a 2026 International Monetary Fund study, they identified ten unique skills most common across Software Development. Only four of these ten skills are in line with programming ability, “Java”, “Javascript”, “Python”, and “SQL” (Jaumotte et al., 2026). The rest of the skills represent higher level concepts such as communication, architecting, and agile methodology.

Programing is simply one of the ways in which a Software Engineer might instruct a computer to perform a specified action. There exists point and click consoles or no code solutions that allow for the provisioning of resources, configurations of security role, and even the creation of jobs using advanced conditional logic all without writing a single line of code. These large language models seem to act as a new method of interaction, allowing for developers to be able to illicit a computer’s action through the interface of plain spoken language, rather than precise and convoluted syntax.

The appropriate incorporation of Artificial Intelligence within both the classroom and the workplace can greatly benefit the user. These systems, when used correctly, are able to relay what could be consistent to high levels of human thought as “tutoring” or other forms of consultation (Wang et al., 2023). Unfortunately it seems as though in most implementations, this beneficial interaction is forewent in favor of productivity over learning. A rich understanding of problem solving is lost when relying on models to complete an entire task. Computational thinking, one of the core philosophies in Computer Science, requires a process containing both “design” and “debugging” (Shute et al., 2017). Both of these skills are non existent when improperly utilizing a tool like AI. Computational thinking is what makes a Computer Science graduate distinct, rather than their ability to code.

3 Conclusion

There is more to Computer Science than simply programming. The ability to effectively decompose, model, communicate, and solve a problem is still an essential skill even in a society dominated by AI. Thinking creatively about a problem and fitting the solution around some given constraints is the true nature of the Computer Science profession. Coding is simply the production of documents existing as the resultant manifestation of computational thinking. Computer Science students often take a data structures and algorithms course, teaching them about elegant solutions to computational problems that might be encountered. Realistically, most individuals will

never have to implement or even interface at all with any of those topics aside from potentially calling predefined implementations. This is one potential outcome for the future of the career Computer Programer within the age of AI not working on critical infrastructure or standard libraries. Programing as a skill might be totally reduced to the occasional peer review, slight debugging, or intelligent prompting rather than as the main focus of work. Artificial Intelligence will certainly bring upon more changes effecting the roles of Computer Scientists, but the value of the Computational Thinking skill is likely to remain.

References

1. Jaumotte, F., Kim, J., Koll, D., Li, E., Li, L., Melina, G., Song, A., & Mendes Tavares, M. (2026). Bridging Skill Gaps for the Future: New Jobs Creation in the AI Age. https://doi.org/10.5089/9798229028196.006.a001

2. Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. Educational Research Review, 22, 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

3. Wang, T., Díaz, D. V., Brown, C., & Chen, Y. (2023). Exploring the role of Ai Assistants in Computer Science Education: Methods, implications, and Instructor Perspectives. 2023 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 92–102. https://doi.org/10.1109/vl-hcc57772.2023.00018