

EDU GUIDE

Why do CS students have to write code by hand?

SUMAIYA RASHID

Evolution is not only anticipated but also required in the realm of programming. The practice of physically writing code on paper appears to be on the verge of becoming obsolete due to the rise of sophisticated code editors and Integrated Development Environments (IDEs).

Handwriting code wasn't seen as antiquated or unusual in the early days of the computing era. Many people now believe that handwriting code is obsolete due to the effectiveness of modern technology. But just as every coin has two sides, this coding approach also has its share of supporters and detractors.

The act of handwriting code pushes programmers to be more deliberate and thoughtful in their approach. Without the benefits of current IDEs, coders are forced to visualise the programme structure, which fosters a deep understanding and an enhanced capacity to handle complicated problems.

Dr Darin Brezeale, Adjunct Instructor at University of Texas at Arlington, says, "The development tools used by programmers have gotten better at potentially helping the programmer by doing some of the work for them. While this helps us do things faster, a downside is that we can rely too much on them and either never learn to do some things ourselves or forget how to do them. Now with ChatGPT, someone can even get a program that sort of does what they want and that they just have to tweak. I foresee there being many 'programmers' who will be unable to actually write any meaningful program completely on their own."

Ayman Azahar, a computer science student at University of Texas at Arlington, expresses, "This approach is inefficient because it significantly slows down the learning curve and limits opportunities for experimentation and exploration. Additionally, small but serious errors are more likely without the automatic error checking provided by most digital coding environments."

Arufa Manar Khanam, a computer science student at Boston University, says, "I was required to write codes on paper throughout all of my coding classes (Java, Python, C). It was mandatory because all of the exams were also on paper. It was challenging because it's hard to see the implementation of the code on paper, as I wasn't able to

test each case."

Thousands of lines of code in complicated, diverse codebases are frequently used in modern development projects. Handwritten codes however, lack the tools for version control, collaboration, and seamless integration – managing such projects through handwritten codes might be a fruitless endeavour.

Arufa adds, "One major disadvantage of handwritten codes is that they do not prepare you for the actual world. If you become accustomed to writing code on paper and practising before coding digitally, you may spend more time troubleshooting issues. Handwritten coding culture will likely drop further as more project-based learning requires you to embed Artificial Intelligence (AI) into your code and/or utilise AI as a tool."

At its heart, the debate over the outdated nature of handwritten codes represents a clash between modernity and tradition. While some believe that adhering to this heritage is equivalent to rejecting development, others maintain that there is intrinsic value in preserving the tradition.

Dr Darin Brezeale, shares, "I teach people to program by writing code in class on a computer. One reason is that it allows me to run the code to demonstrate what the output of the program is. I only ask students to write code on paper during exams. Even if I didn't have to worry about the problems with giving someone Internet access during an exam, writing programs by hand shows what the student knows about programming syntax and how well they can envision the logical steps necessary to perform a particular task."

Stating that handwriting code is categorically outdated may be oversimplified. The methodical, leisurely pace of writing programmes by hand serves as a subtle reminder of the depth and value that lie in the process of creation in a society that is fascinated by the appeal of speed and efficiency.

Dr Darin further emphasises, "Some people, not always beginners, don't have the mental models that represent a task and the sequence of programming steps necessary to perform the task. For them programming is a trial-and-error process. Type something in, see what happens, type something else, see what happens, and so forth until they think the program does what they want. I think that these people can't write complete programs on paper because they can't see the sequence of programming logic necessary to perform the task."

It is essential to recognise that handwriting codes and contemporary coding techniques can coexist. For instance, contemporary coding environments will continue to be the staple in professional settings while handwriting codes can find its place in educational settings.

However, it is detrimental to only continue the practice of handwritten coding since it prevents students from being familiar with the modern tools and technologies that are now accepted as industry standards.

Sumaiya Rashid is a student at Independent University, Bangladesh.



DO'S & DON'TS OF A COMPUTER SCIENCE STUDENT

THE DO'S:

- Be punctual when it comes to lab assignments. The labs will help you implement your knowledge through problem solving.
- Solve problems and attend programming contests.
- Seek help from your peers and faculty members instead of just copy-pasting someone else's work.
- Make all the mistakes you want but at the same time, you need to know what it takes to rectify them, especially when it comes to coding problems.

THE DON'TS:

- Plagiarism is never the solution when it comes to coding. Instead, try to solve the problems by yourself. Take help from others if necessary.
- Avoid trying to learn more than one programming language at a time.
- Don't just rely on class lectures if you want to develop your coding skills.



ILLUSTRATION:
FATIMA JAHAN ENA