

Nvidia CEO Jensen Huang recently said that AI will obviate the need to learn programming languages, but not everyone agrees

Coding is still relevant IN THE AGE OF AI

Akhil.George@timesgroup.com

Jensen Huang, the leather jacket-toting CEO of chip behemoth Nvidia, recently made global headlines when he asserted that programming languages are quickly becoming a thing of the past and that human language prompts will replace complex programming languages like C++ and Python. Jensen argued that kids today should instead aim to become domain experts – like in finance or manufacturing.

“Over the last 10-15 years, almost everybody who sits on a stage like this would tell you that it is vital that your children learn computer science, everybody should learn how to program. In fact, it is almost exactly the opposite. It is our job to create computing technology such that nobody has to program, and that the programming language is human. Everybody in the world is now a programmer. This is the miracle of AI,” he said at the recent World Government Summit in Dubai.

His remarks caused quite a debate within the tech community, so we asked a few experts in the field for their views.

Deepak Visweswaraiyah, VP of Platform Engineering & site MD at Pegasystems India, says that while software development is evolving with the onset of automation tools, AI integration, and low-code development, what it really means is that software developers now have new tools helping them with accelerating code completion, optimisation, and testing. “But, despite these sleek tools, the expertise that software developers bring in is necessary for maintenance, troubleshooting, and upgrading.” Ajay Kashyap, chief product officer at LEAD Group, a school edtech unicorn, says that suggesting that children need not study programming languages at all and should instead only focus on becoming domain experts overlooks

FUTURE OF LEARNING

the fundamental importance of understanding the principles of programming – it provides a framework for logical thinking, problem-solving, and creativity, all of which are crucial 21st-century skills.

Even if many of the experts are wrong and AI tools become more than adept at writing all the code we need in the future, it still does not change the fact that developing software is a lot more than just coding.

“Developing software requires several human attributes, such as understanding user needs, designing systems, testing, and maintenance. As the field evolves, particularly with AI’s growing role, viewing software development as an ever-evolving discipline is essential,” says Nishant Patel, founder and CTO of Contentstack.

Jhilmil Kochar, MD at CrowdStrike India, says that programming skills remain highly relevant even with the rise of AI. “Logical thinking, problem solving, innovation and design are all critical skills for programmers that AI cannot fully replace. Even as their roles evolve, software engineers are still needed to define problems, which is a first step towards an elegant solution, to understand domains, and design systems that are simple, user-friendly, robust and secure.”

A programmer who is not innovative and a problem solver will find it difficult to grow, she adds. “Today, companies are also hiring software engineers with AI experience. A world where AI solves every software problem is still a myth. AI can be a very powerful tool to aid productivity and creativity but is not a replacement for humankind.”

One thing that all the experts seem

Computer science enthusiasts should strive towards understanding of core concepts and pursuing well rounded learning experiences. Alongside this, proficiency in common programming languages like Java, C/C++, and scripting languages such as PHP, Python or Perl is crucial. These skills not only enhance the ability to detect potential attacks on systems but also facilitate the development of effective countermeasures.



Jhilmil Kochar | MANAGING DIRECTOR, CROWDSTRIKE INDIA

Rather than replacing the need for programming knowledge, AI is shifting the focus towards leveraging these technologies to complement domain expertise and enhance problem solving skills. This paradigm shift supports our belief in the convergence of digital technology and traditional engineering, fostering a new generation of engineers who are adept at both.



Santosh Singh | GLOBAL HEAD OF MARKETING, BUSINESS EXCELLENCE & INNOVATION, TATA TECHNOLOGIES

While mastering AI & ML, Python, R, and SQL, students today in computer science also need to be aware of privacy, bias, fairness, transparency, and accountability. Understanding deep learning basics and data management is crucial for success.



Beyond coding, they should develop a diverse skill set, blending technical expertise, critical thinking, creativity, adaptability, and a growth mindset.

Deepak Visweswaraiyah | VP, PLATFORM ENGINEERING, AND SITE MANAGING DIRECTOR, PEGASYSTEMS, INDIA

to agree on is the growing importance of domain expertise, since it’s remarkably difficult to build complex software solutions for industries that one doesn’t have any knowledge of.

“Domain knowledge is crucial for developers to align their coding decisions with the users’ unique challenges, requirements, and intricacies. This knowledge enhances industry collaboration, communication, and problem-solving, fostering more effective and tailored solutions. Overall, domain knowledge empowers developers to create applications that are not just technically sound but also aligned with broader strategic goals,” says Muthumari S, global head of data and AI studio at Brillio.

WHAT TO STUDY

Revathi Srinivasan, principal of Smt Sulochanadevi Singhania School, says that students should prioritise studying a broad range of skills within the computer science field, including algorithm design, system architecture, user experience design and project management. “Additionally, fostering domain expertise along with programming skills can enhance students’ abilities to apply technology effectively in various industries and domains.”

Farun Gulyani, head of AI practice at Newgen Software, says that if stu-

While I would caution and call it premature to dismiss programming education altogether, the growing capabilities of AI in automating coding tasks suggest a shift. AI can now assist with coding but understanding programming fundamentals remains essential for leveraging AI tools effectively and solving complex problems. The future likely will not see AI replacing programmers but rather will augment their work, making domain expertise increasingly important.



Nishant Patel | FOUNDER & CTO, CONTENTSTACK

Just because we have calculators, we haven’t given up on teaching elementary math to students in their early years. Likewise, just because we have Gen AI doesn’t mean that coding skills can be dispensed completely. I anticipate that those who learn to use Gen AI as an assistant will succeed, versus those who see it as a master who knows it all.



Vinay Rai | EXECUTIVE VP – ENGINEERING, NETRADYNE TECHNOLOGY

While AI can assist with code generation, it is unlikely to entirely take over coding. AI enables our best programmers to amplify manifold their programming skills, and prowess by automating repetitive workflows, removing the need to learn syntax, and scaling productivity through collaborative learning and reusability across developers. More citizen developers with domain knowledge may well be getting hired today, but the need for the developer community remains unchanged.



Muthumari S | GLOBAL HEAD, DATA & AI STUDIO, BRILLIO

dents want to work at large global tech companies that build enterprise level software, then it is essential for computer science students to comprehend the advantages and disadvantages of different programming languages, as well as how to effectively utilise them. “Understanding fundamental concepts such as when to employ compiled-time languages versus interpreted languages is crucial. This necessitates a deep understanding of programming languages, data structures, and their low-level utilisation when interfacing with hardware instructions.”

Contentstack’s Nishant says students should prioritise a blend of skills: computational thinking, software engineering principles, user experience design, and project management, alongside domain-specific knowledge. “Proficiency in data structures, algorithms, and at least one programming language remains foundational.”

Turning to the question of teaching young children to code, CrowdStrike’s Jhilmil says it is a nuanced issue and more complex. “Programming promotes logical thinking and creativity, which are beneficial skills. However, the world being as competitive as it is, in the drive to equip kids with technical skills early on, their overall development and childhood experience must not be compromised.”