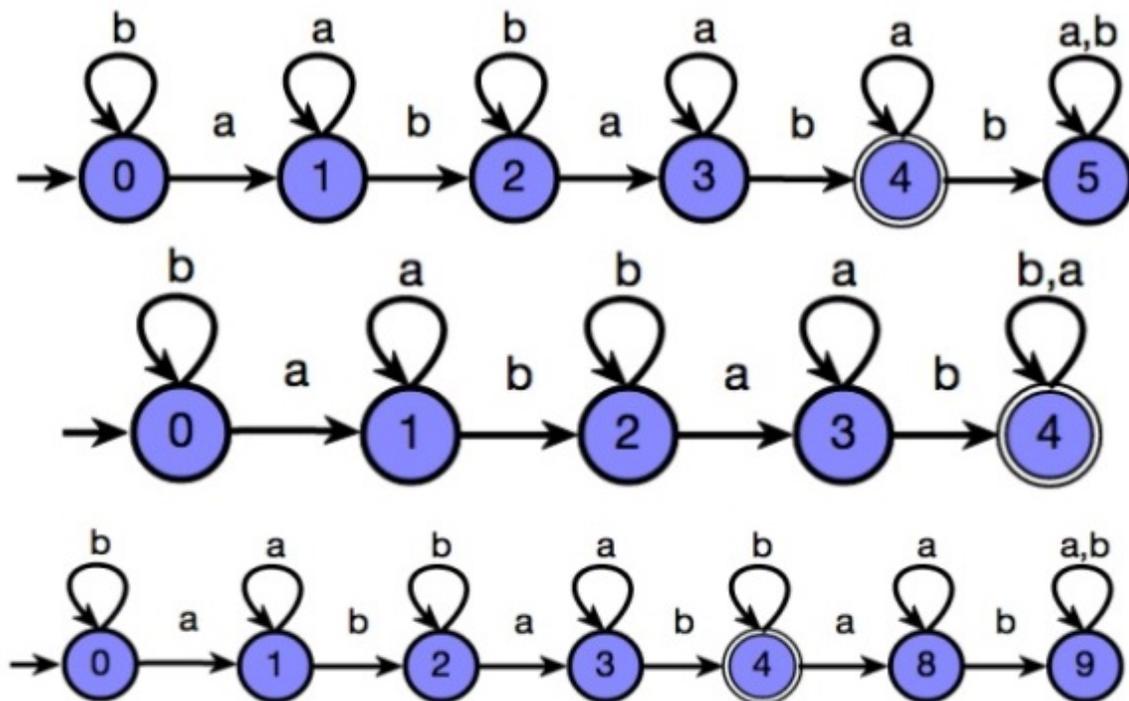


Computer programming made easier

August 12 2016



Automata Tutor, a tool developed by the NSF-funded Expeditions in Computer Augmented Program Engineering project, automatically grades student answers to a programming problem. Credit: ExCAPE

Nobody said computer programming was easy. But maybe in the future, it could be.

In order to simplify program development, a National Science Foundation (NSF)-supported project called Expeditions in Computer Augmented Program Engineering (ExCAPE), is developing technology that provides human operators with automated assistance.

"Computers have revolutionized our daily lives, and yet the way we program computers has changed little in the last several decades," said Rajeev Alur, a professor in the department of computer and information science at the University of Pennsylvania.

Alur heads a team of researchers—representing nine leading computer science programs in the U.S.—that collaborates on the ExCAPE project. NSF supports ExCAPE with a \$10 million, five-year Expeditions in Computing award, which funds interdisciplinary research teams working to transform computing and technology.

Alur said the team is taking on a longstanding problem: "Software development remains a tedious and error-prone activity."

Using a model of programming called automated program synthesis, however, computers can generate pieces of code based on a user's intent, expressed using various non-code-based forms, such as examples, demonstrations or natural language commands.

"ExCAPE aims to change programming from a purely manual task to one in which a programmer and an automated program synthesis tool can collaborate to generate software that meets its specification," Alur said.

By removing the need for would-be programmers to learn esoteric programming languages, the method has the potential to significantly expand the number of people engaged in programming in a variety of disciplines, from personalized education to robotics.

Programming tools

Emerging technology known as Software-Defined Networks (SDN) allows network operators to tailor a computer network to the traffic running on it, thereby improving efficiency. Most network operators, however, are not traditional programmers and, as a result, cannot take full advantage of all the technology offers.

To address this shortcoming, the ExCAPE team developed a tool called NetEgg that lets a network operator specify the desired functionality of a switch using examples. NetEgg then automatically generates the code needed to implement that behavior while ensuring maximal throughput for network traffic.

Now patented, NetEgg has already been tested in a classroom setting and forms the basis of an NSF I-Corps project, which will explore the product's transition to commercial deployment.

Computer-aided education and beyond

Looking at the growing area of online learning, the ExCAPE team further recognized the role that program synthesis tools could play in generating automatic feedback for students—analyzing their solutions, grading their assignments, and providing meaningful explanations of their mistakes.

That's why the team created Automata Tutor, which has been used by more than 5,000 students from more than 10 universities around the world. Alur and his colleagues presented the results from the early deployment of Automata Tutor in ACM Transactions on Computer-Human Interaction and at the International Joint Conference on Artificial Intelligence.

The group has created other tools, including AutoProf, which provides feedback on introductory programming assignments in computer languages, such as Python. Another tool, CPSgrader, automatically grades laboratory courses in cyber-physical systems and provides feedback.

More broadly, the ExCAPE team was able to develop a method that formalized and standardized the core computational problem in emerging synthesis tools. Called Syntax-Guided Synthesis, the new method has allowed the team to build a number of prototype solvers over the past two years.

"This effort has been instrumental in advancing the state-of-the-art in computational approaches, and it has facilitated novel applications of program synthesis, for instance, in automatic optimization of programs for quantum computers," Alur said.

Industry adoption

The ExCAPE team's research has affected the commercial software world, too. Its notion of syntax-guided synthesis inspired Microsoft to create automated program synthesizers for its suite of software.

"At Microsoft, we have invested significantly in the field of program synthesis, especially programming-by-examples, and with applications to end-user programming," said Sumit Gulwani, of Microsoft Research, USA.

Microsoft started out by developing domain-specific synthesizers such as FlashFill and FlashExtract, each of which uses examples to generate custom code that improves efficiency. FlashFill, which was released as a feature of Microsoft's Excel 2013, allows data entered into one column of a worksheet table to be entered in a new table column using only a

few keystrokes. FlashExtract, which was included in Microsoft's PowerShell and Operations Management Suite, extracts structured data from semi-structured log files using examples.

The technology giant has also developed a generic programming-by-example synthesizer called FlashMeta.

"All of our ongoing development of by-example synthesizers at Microsoft for various domains is now being carried out over the FlashMeta framework," Gulwani said. "In fact, we have set up an entire research and engineering team for development of this framework, called PROSE. This has yielded one order of magnitude effectiveness in the overall development process."

In years to come, the process of using coding languages for programming may be seen as an evolutionary step in computing, just as other methods replaced the punch cards and assembly languages used to program early computers.

"This project builds on decades of foundational advances in formal methods and programming languages," says Nina Amla, program director in the Division of Computing and Communication Foundations at NSF. "It signals a paradigm shift in the way we teach basic programming principles, and develop reliable software systems."

Provided by National Science Foundation

Citation: Computer programming made easier (2016, August 12) retrieved 26 April 2024 from <https://phys.org/news/2016-08-easier.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is

provided for information purposes only.