EVERYONE IS A PROGRAMMER:

MAKING THE WORLD A CONTROL SYSTEM

when everything is programmable: LIFE IN A COMPUTATIONAL AGE

As people begin to live in and interact with a world in which everything is programmable, the need will grow for tools and institutions that teach how to think computationally. As the amount of data we have at our disposal increases exponentially, more and more disciplines will require computational thinking skills in order to make sense of this information. Novice-friendly programming languages, early education in algorithmic thinking and logic, and technologies that teach the fundamentals of programming virtual and physical worlds will enable us to manipulate our built environments and enhance our interactions with the different programmable domains of our lives.

LEARNING TO DEAL WITH DATA

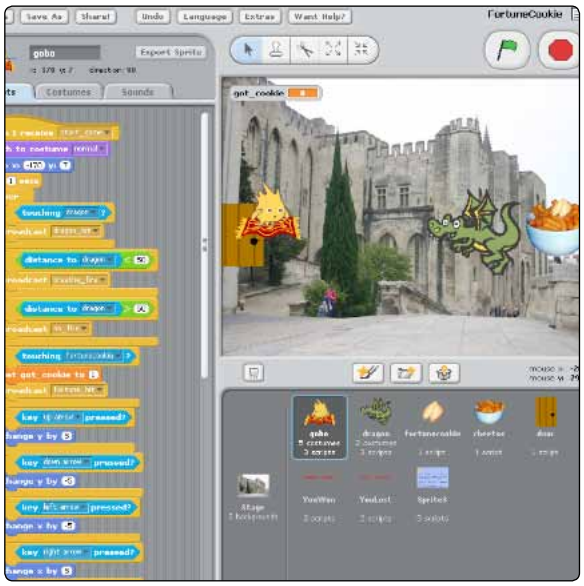
A growing number of disciplines are becoming computationally driven. As leading-edge research in areas as diverse as political science and biology begin to profit from the continuous production of massive data sets, researchers will need to get better at abstracting out problems and applying computational and logical thinking to those problems. There will be a shift toward data analysis over traditional types of experimentation as large-scale research programs such as the Large Hadron Collider begin to produce staggering amounts of data and as models of social systems in virtual worlds become more sophisticated. In turn, hypotheses will be heavily informed and refined by the ability to generate highly precise models and run sophisticated statistical analyses on previously qualitative processes.

COMPUTATIONAL THINKING AS THE NEW MATH

Changes to the school curriculum will begin to reflect the growing social need for people who are familiar with computational thinking. The institutionalization of "computer thinking" education will provide the foundation for citizens of a programmable world to learn the higher-order skills—problem solving, creative thinking, logical reasoning, and systematic experimentation—that will allow them to meaningfully manipulate the world around them. As computational thinking becomes embedded in how we understand our lives and our environment, we will see a shift in how subjects like the life sciences are taught. Building on the scientific method, in which we are taught how to understand our natural world, students will be taught how to change the world around them.

FULFILLING THE PROMISE OF DIGITAL LEARNING

Prophesies of remote learning that are indistinguishable from a traditional education have been slow to be fulfilled. However, as a result of a combination of institutional support for providing access to education materials, pervasive high-speed Internet, and functioning community support structures, programming education will dismantle the remaining barriers to e-learning. Capitalizing on the desire to teach computational thinking, computer science education is setting up structures for digital education that will make it a truly viable option in a variety of fields, and is experimenting with new ways of connecting teachers and students that will ensure that future learners experience an education that is interactive and tailored to their specific needs, level of knowledge, pace of learning, and learning style.



Scratch, "designed to help young people (ages 8 and up) develop 21st century learning skills"

Simulation:
Modeling possibility space

Cloud Computing:
Supercomputing on demand

Molecular Engineering:
Building from the bottom up

Pervasive Wireless:
Continuous connection

Sensors and Sensor Networks:
Everything in its right place

Signals:

NSF CLOUD COMPUTING RESEARCH GRANTS
(LEARNING TO DEAL WITH DATA)

In recognition of the growing need for large-scale computational analysis across a range of disciplines, the National Science Foundation has begun to fund education efforts that aim to "explore innovative research ideas in data-intensive computing." In April 2009, 14 universities received approximately $5 million in grants aimed at advancing computational expertise for application in fields as disparate as disaster mitigation, environmental protection, astrophysics, oceanography, and analysis of information flow through social networks.

Source: http://www.nsf.gov/cise/clue/index.jsp

CARNEGIE MELLON CENTER FOR COMPUTATIONAL THINKING
(COMPUTATIONAL THINKING AS THE NEW MATH)

Carnegie Mellon's Center for Computational Thinking leads efforts to teach the tenets of computational thinking and encourages its application to nontraditional domains. The center's major activity is PROBEs (PROBlem-oriented Explorations), thought experiments that challenge participants to harness computational thinking to approach problems not traditionally solved via computation. For example, one PROBE asked, "What would happen if surgeons and medical clinics used computational thinking in order to make organ transplantation decisions? Is it possible to optimize the allocation of organs so that many more people can be saved?"

Source:http://www.cs.cmu.edu/~CompThink/index.html

CS UNPLUGGED (COMPUTATIONAL THINKING AS THE NEW MATH)

Developed by the University of Canterbury in Christchurch, New Zealand, and sponsored by Google and Carnegie Mellon's CS4HS (Computer Science for High School Teachers), CS Unplugged uses noncomputer methods and materials such as "games and puzzles that use cards, string, crayons, and lots of running around" to teach the fundamental concepts and tenets of computer science without requiring a computer interface. The program demonstrates that computational approaches to the world can be taught to children at a much earlier age and in a more enjoyable, more familiar manner than previously thought.

Source: http://www.csunplugged.org

SCRATCH (FULFILLING THE PROMISE OF DIGITAL LEARNING)

Scratch is an "interactive learning environment" (ILE) that aims to foster immediate engagement as it helps teach the fundamentals of programming to children. Scratch is fostering the skills needed to understand and apply higher-level programming languages, while simultaneously introducing the collaborative sense of community that is the norm in the programming world. Other examples of ILEs include Alice (alice.org) and Greenfoot (greenfoot.org).

Source: http://scratch.mit.edu/

What difference does this make?

When computational thinking becomes the foundation of educational curricula people's expectations of the world and employers' expectations of employees will change. Education in ethics will be needed alongside education in computational thinking.

PROGRAMMING EDUCATION BY COMPUTATIONAL THINKING

Manipulating and modifying will become central to the school curriculum. While not everyone will be able to (or need to) master higher-level programming languages, the increase in a worldwide population that is familiar with malleable environments will result in a cascade of changes in how traditionally noncomputational subjects are taught.

PROGRAMMING EVERYDAY EXPECTATIONS

As more of the world becomes programmable, customizable, and modifiable, and as more people are able to program the world, expectations of interactivity, customization, and control will grow. After habituating to a world in which many interactions can be programmed, people will begin to demand more reactivity in the products they interact with, even where programmed interactivity is not enabled.

APPLYING COMPUTATIONAL SKILLS BECOMES MAINSTREAM

Currently, HR departments value applicants who are familiar with basic applications, such the Microsoft Office suite. Soon, we may see a shift in the expectations surrounding hiring new employees as employers begin to look for resumes that mention "familiarity with programmable systems" and "proficiency in environmental manipulation."

NEED FOR NEW ETHICS EDUCATION

With increased understanding of how we can alter the fundamental building blocks of life, students will also need to learn how to ethically and responsibly affect changes in the previously immutable biological design of organisms. This education will be essential to ensuring that citizens of a world in which everything is programmable will make changes that are for the betterment of society and the natural world.

What to do differently?

An organization operating in a programmable world will require systemwide embedded computational thinking among its employees. At the same time, in order to remain competitive in their jobs, employees will need to understand how to make use of the programming tools at their disposal.

HOST COMPUTATIONAL POTLUCKS

Holding employee training and familiarization sessions on programming tools is a way to educate your organization's members in computational thinking. By hosting seminars that emulate programs such as Carnegie Mellon's PROBEs, your organization can help employees learn how to apply computational thinking to their work-specific tasks and technologies.

ENCOURAGE EMPLOYEE EXPERIMENTATION IN ILES

Interactive learning environments such as Alice and Scratch are excellent venues for experimenting with the fundamentals of programmatic thought and learning how to apply computational methodology in a fun, low-risk environment. Hosting organization-wide competitions for the most enjoyable, most artistic, and most advanced beginner programs is a great way to encourage employees to learn more about computational thinking.

CREATE PRODUCTS AND SERVICES THAT ARE MORE CUSTOMIZED

Customers are going to expect the goods and services they buy to be modifiable, customizable, and otherwise responsive to their individual needs and desires. They will look for goods and services that are configurable by the end user so they can inject their own personalities into the products they use. Companies that offer such goods and services will benefit from an increase in consumer brand loyalty.

INVEST IN LIGHTWEIGHT IT

Computationally thinking employees who understand how to combine and reconfigure different modules of functionality can help decrease your company's IT costs. Adopting a modular, lightweight IT system harnesses individual knowledge in order to solve individual problems and allows employees to configure and reconfigure their applications in response to changes in their individual needs.

INSTITUTE FOR THE FUTURE

Technology Horizons Program
124 University Avenue
2nd Floor
Palo Alto, CA 94301
t 650.854.6322
f 650.854.7850
www.iftf.org