

## CS10K, CS Principles e o Computing Curricula 2013

Conheça três projetos que estão em andamento nos EUA sobre Educação em Computação

John Forman, jforman@nac.softex.br, SOFTEX  
Mirella M. Moro, mirella@dcc.ufmg.br, UFMG

*Este artigo apresenta um breve relato sobre três novos projetos que estão em andamento nos EUA: o CS10K para habilitar professores do ensino médio/fundamental a ensinar Computação nas escolas; o CS Principles para definir um curso introdutório em Computação nacional; e o Computing Curricula 2013, para definir o novo currículo da Computação da ACM/IEEE.*

### CS10K: 10 thousand schools with 10 thousand well prepared CS teachers by 2015

As estatísticas americanas mostram que em 2018 existirão 1,4 milhões de ofertas de trabalho para especialistas em computação. Até lá, as universidades americanas terão formado recursos humanos para cobrir um terço dessas vagas. Os resultados do relatório “*Running on Empty: The Failure to Teach K-12 Computer Science in the Digital Age*”, lançado em um evento em Washington, DC, no ano de 2010 também trouxe resultados nada animadores (veja Recursos). Os desafios incluem a grande falta de interesse em computação, o envolvimento escasso das minorias na área (incluindo mulheres) e a presença negligenciada da Computação no ensino médio/fundamental (computação acadêmica não é ensinada e a nossa comunidade simplesmente não se envolve com ensino que não seja superior).

Apesar de já existirem projetos conhecidos buscando melhorar o ensino de Computação nos EUA, todos são realizados regionalmente, não existindo ainda um esforço/projeto nacional. Entre os projetos estão o modelo de currículo para Ciência da Computação no ensino médio, produzido pela ACM e que serve de guia para as secretarias estaduais de educação desenvolver iniciativas em seus respectivos estados (a última revisão do modelo foi feita em 2003, veja em Recursos). Outro projeto é o *Computing in the Core* que visa melhorar o nível do ensino de Ciência da Computação, iniciativa esta considerada um dos pilares do CS10K.

O CS10K é um projeto colaborativo em larga escala para ajustar a escala, o currículo e a pedagogia relacionados com o ensino de Ciência da Computação em todos os níveis de ensino, mas priorizando as escolas do nível médio (*high schools*) e os cursos de introdução à Computação no ensino superior. Este projeto busca aproximar um número significativo de colaboradores, incluindo agências de governo, comunidade acadêmica, fundações e iniciativa privada. **O objetivo é atingir a meta de ter 10.000 professores em 10.000 escolas ensinando um novo currículo de computação até o ano de 2015.**

Um componente do CS10K é o projeto *CS Principles*, que é explicado na próxima seção. É importante notar que a revisão do currículo de Computação no ensino médio não é o maior desafio do projeto, e sim conseguir alcançar o número de 10.000 professores de Computação em 10.000 escolas. Muito poucos são os professores no ensino médio nos EUA com uma formação em Ciência da Computação. Pensando nisso, o projeto irá lançar um programa de preparação de professores em larga escala, utilizando diferentes abordagens para atingir o maior número possível de professores. A idéia é usar pesadamente material de apoio na web, redes sociais e outras tecnologias, sem excluir treinamentos presenciais. Uma alternativa é convocar universidades e centros de pesquisa para se envolver diretamente neste esforço de treinamento de professores.

O CS10K poderia ser um esforço até internacional, na medida em que a carência de profissionais na área de TI é um fenômeno que vem acontecendo em diferentes partes do mundo. Será que conseguiríamos fazer o equivalente (guardadas as devidas proporções) no Brasil??

## CS Principles

CS Principles em um segundo:

- O que: introdução alternativa, rigorosa e criativa à Ciência da Computação;
- Quem: comunidade colaborativa, incluindo academia, College Board e NSF;
- Quando: agora.

Este projeto consiste na elaboração de um novo programa de Computação para o nível médio com foco na oferta de disciplinas de *Advanced Placement* – mecanismo existente nos EUA onde determinadas disciplinas do ensino médio podem ser cursadas e já valem créditos para cursos universitários. O problema é que tal ajuste nas disciplinas oferecidas no nível médio exige também a atualização dos cursos introdutórios de Ciência da Computação no nível superior. Resultados piloto dessa iniciativa já estão disponíveis no site do projeto. Estão também abertas as inscrições para instituições interessadas em fazer parte de uma segunda etapa de testes piloto na oferta da disciplina utilizando o currículo revisado proposto pelo projeto. As linhas mestras para desenvolver um currículo de computação levaram em consideração as sete grandes ideias (*7 Big Ideas*) apresentadas na Tabela 1.

**Tabela 1. Seven Big Ideas (foi mantido em inglês)**

<b>Idea</b>	<b>Explanation summary</b>
<b>1. Computing is a creative human activity that engenders innovation and promotes exploration</b>	Creativity and computing are prominent forces in innovation; the innovations enabled by computing have had and will continue to have far-reaching impact. At the same time, computing and computer science facilitate exploration and the creation of knowledge. This course will emphasize the creative aspects of computing. Students in this course will create interesting and relevant artifacts with the tools and techniques of computing and computer science.
<b>2. Abstraction reduces information and detail to focus on concepts relevant to understanding and solving problems</b>	People use abstraction every day, but it is a central problem-solving technique in computer science. Traditional examples of abstraction in computer science include control and data abstraction in programming languages as well as analyzing and understanding hardware and software systems. This course will include examples of abstractions used in modeling the world, in managing complexity, and in communicating with people as well as with machines. Students in this course will learn to work with multiple levels of abstraction while engaging with computational problems and systems.
<b>3. Data and information facilitate the creation of knowledge</b>	Computing enables and empowers new methods of information processing that have led to monumental change across disciplines, from art to business to science. A staggeringly large amount of raw data provides part of the foundation of our information society and economy. People use computers and computation to translate, process, and visualize raw data, creating information. Computation and computer science facilitate and enable a new understanding of data and information that contributes knowledge to the world. Students in this course will work with data using a variety of tools and techniques to better understand the many ways in which data is transformed into information and knowledge.
<b>4. Algorithms are tools for developing and expressing solutions to computational problems.</b>	Algorithms help elementary school students learn to multiply, but algorithms realized in software have affected the world in profound and lasting ways. The development, use, and analysis of algorithms is one of the most fundamental aspects of computing. Students in this course will work with algorithms in many ways: they will develop and express original algorithms, they will implement algorithms in some language, and they will analyze algorithms both analytically and empirically.
<b>5. Programming is a creative process that produces computational artifacts</b>	Programming and the creation of software have changed our lives. Programming results in the creation of software, but it facilitates the creation of more general computational artifacts including music, images, visualizations, and more. In this course programming will enable exploration and study as well as being the object of study. This course will introduce students to the concepts and techniques used in writing programs and to the ways in which programs are developed and used by people; the focus of the course is not programming per se, but on all aspects of computation. Students in this course will create programs, translating human intention into computational artifacts.
<b>6. Digital devices, systems, and the networks that interconnect them enable and foster computational approaches to solving problems</b>	Digital devices and the Internet have had a profound impact on society. The principles of systems and networks that helped enable the Internet are also critical in the implementation of computational solutions. Computer networks support communication and collaboration. Students in this course will gain insight into how systems and networks operate, to the principles that facilitate their design, and will analyze the effects of systems and networks on people and society.
<b>7. Computing enables innovation in other fields including science, social science, humanities, arts, medicine, engineering business</b>	Computation has changed the way people think, work, live, and play. Our methods for communicating, collaborating, and problem-solving, and doing business have changed and are changing due to innovations enabled by computing. Many innovations in other fields are fostered by advances in computing. Computational approaches lead to new understandings, new discoveries, and new disciplines. Students in this course will become familiar with the many ways in which computing enables innovation in other fields.

Além disso, o currículo proposto deve permitir desenvolver os pontos do *Six Computational Thinking Practices*:

1. *Analyzing effects of computation*
2. *Creating computational artifacts*
3. *Using abstractions and models*
4. *Analyzing problems and artifacts*
5. *Communicating processes and results*
6. *Working effectively in teams*

### Computing Curricula 2013: Computer Science

Já são tradicionais os esforços para estabelecer um currículo de Computação que possa servir de orientação para programas de graduação em Ciência da Computação (e demais cursos) em nível internacional. Como a tecnologia evolui rapidamente, de tempos em tempos o currículo de referência precisa ser revisado. O processo de revisão da versão atual do currículo para Ciência da Computação foi iniciado em 2010. As questões a serem resolvidas em tal atualização são relacionadas não apenas com a *evolução* da própria Computação, mas também com o *uso* da Computação em outros campos do conhecimento. A solução exige o envolvimento de um número maior de pessoas no processo, que precisa também considerar necessidades regionais e diferenças culturais.

O currículo de referência atualmente vigente foi publicado em 2001, seguido de uma revisão em 2008 que apenas atualizou alguns tópicos não abordados na versão original e não enfrentou questões mais estruturais que já começavam a ser debatidas naquela ocasião. A nova versão a ser publicada em 2013 pretende ser de fato uma reflexão aprofundada do currículo como um todo, redefinindo unidades de conhecimento, revisando os conceitos básicos essenciais e buscando exemplos concretos de cursos e currículos que atendam as futuras recomendações. Norteiam o trabalho sendo desenvolvido:

- O papel da Ciência da Computação como componente fundamental para outras áreas do conhecimento, entendendo a natureza multidisciplinar do uso da computação, sem no entanto perder a identidade dos conhecimentos que constituem o núcleo desta área do conhecimento;
- O tamanho do currículo, já que a Computação vem se desenvolvendo mas não é possível aumentar a carga horária dos cursos da área. Assim como o currículo de 2001 reduziu a carga horária sugerida em comparação com o currículo de 1991, a versão de 2013 deve também diminuir a carga horária sugerida para disciplinas obrigatórias, sugerindo a oferta de cursos mais flexíveis, sem perder a essência de uma educação formal em Computação;
- A melhor compreensão do movimento que defende o *Computational Thinking*, analisando o impacto disto no currículo, bem como em audiências não necessariamente ligadas a cursos de Computação;
- A melhor definição de disciplinas e conteúdos. O currículo de 2001 oferece descrições genéricas de blocos de conhecimento que devem ser abordados, sem definir claramente disciplinas e respectivos conteúdos. Essa abordagem se mostrou pouco eficaz, e a edição 2013 pretende trazer exemplos concretos de cursos e disciplinas aplicados com sucesso em diferentes instituições, funcionando como referências práticas (mas não mandatórias) de como as recomendações podem ser adotadas;
- As características institucionais, já que Computação é ensinada por diferentes tipos de instituições, variando a carga horária e o tempo em que os alunos permanecem nos respectivos cursos. Assim, o currículo de 2013 deve oferecer diferentes modelos para atender tanto o nível médio e técnico, como também o ensino superior e as necessidades da indústria;
- A incorporação de novas linhas de tecnologia. Aqui, despontam a computação paralela e a necessidade de melhor entender segurança de computadores e redes.

## Conclusão

Este artigo apresentou três grandes iniciativas americanas para a melhoria da educação em Computação tanto nas escolas quanto no ensino superior. Embora a realidade nos EUA seja completamente diferente do Brasil, tanto em termos econômicos quanto em educação, essas iniciativas mostram a preocupação daquele país em relação à área de Computação. É importante notar que a diminuição no interesse dos estudantes de ensino médio em cursar Computação não é um problema específico dos EUA, visto que também aflige o Brasil. Mesmo que nosso país não tenha as mesmas condições para, por exemplo, qualificar 10 mil professores em poucos anos, as iniciativas americanas podem e devem nos servir de inspiração para atacar o mesmo problema. Desse modo, esperamos que este artigo acenda alguma fagulha de interesse na comunidade para que, muito em breve, tenhamos força para colocar fogo e amenizar (senão resolver) este problema.

## Recursos

ACM Currículo Computação Ensino Médio <http://csta.acm.org/Curriculum/sub/CurrFiles/K-12ModelCurr2ndEd.pdf>

Computing in the Core <http://www.computinginthecore.org>

CS Principles <http://www.csprinciples.org>

CS10K <http://www.computingportal.org/cs10k>

CS10K artigo no SIGCSE'2011 <http://portal.acm.org/citation.cfm?id=1953193>

NSF CE21: Computing Education for the 21<sup>st</sup> Century [http://www.nsf.gov/funding/pgm\\_summ.jsp?pims\\_id=503582](http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503582)

Running on Empty <http://csta.acm.org/runningonempty>

## Sobre os autores



John L. Forman possui graduação em Engenharia de Computação pela PUC-Rio (1987), graduação em Tecnólogo em Processamento de Dados pela PUC-Rio (1988), mestrado em Informática (Engenharia de Software) pela PUC-Rio (1992) e Pós-graduação em Gestão Empresarial pela COPPEAD/UFRJ (1995). Atualmente é diretor de Capacitação & Inovação da Sociedade Softex, Presidente do conselho deliberativo da Riosot (Agente Softex do Rio de Janeiro), membro do Conselho Consultivo da Assespro-RJ e sócio-titular da Sociedade Brasileira de Informática em Saúde. Tem experiência na área de Ciência da Computação, com ênfase em Engenharia de Software, atuando principalmente nos seguintes temas: saúde, educação, integração corporativa e convergência digital.



Mirella M. Moro é professora adjunta no Departamento de Ciência da Computação da Universidade Federal de Minas Gerais - UFMG. Possui doutorado em Ciência da Computação pela University of California in Riverside (2007), e graduação e mestrado em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (UFRGS). É Diretora de Educação da SBC e editora-chefe da SBC Horizontes.