Should My Kid Learn to Code?

Written by Maggie Johnson, Director of Education and University Relations, Google

Over the last few years, successful marketing campaigns such as Hour of Code andMade with Code have helped K12 students become increasingly aware of the power and relevance of computer programming across all fields. In addition, there has beengrowth in developer bootcamps, online "learn to code" programs (code.org, CS First,Khan Academy, Codecademy, Blockly Games, etc.), and non-profits focused specifically on girls and underrepresented minorities (URMs) (Technovation, Girls who Code, Black Girls Code, #YesWeCode, etc.).

This is good news, as we need many more computing professionals than are currently graduating from Computer Science (CS) and Information Technology (IT) programs. There is evidence that students are starting to respond positively too, given undergraduate departments are experiencing capacity issues in accommodating all the students who want to study CS.

Most educators agree that basic application and internet skills (typing, word processing, spreadsheets, web literacy and safety, etc.) are fundamental, and thus, "digital literacy" is a part of K12 curriculum. But is coding now a fundamental literacy, like reading or writing, that all K12 students need to learn as well?

In order to gain a deeper understanding of the devices and applications they use everyday, it's important for all students to try coding. In doing so, this also has the positive effect of inspiring more potential future programmers. Furthermore, there are a set of relevant skills, often consolidated as "computational thinking", that are becoming more important for all students, given the growth in the use of computers, algorithms and data in many fields. These include:

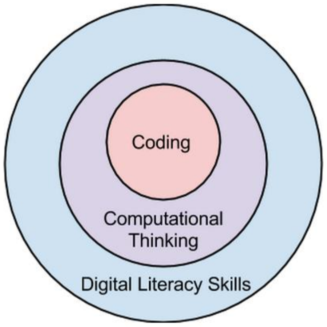
- Abstraction, which is the replacement of a complex real-world situation with a simple model within which we can solve problems. CS is the science of abstraction: creating the right model for a problem, representing it in a computer, and then devising appropriate automated techniques to solve the problem within the model. A spreadsheet is an abstraction of an accountant's worksheet; a word processor is an abstraction of a typewriter; a game like Civilization is an abstraction of history.
- An algorithm is a procedure for solving a problem in a finite number of steps that can involve repetition of operations, or branching to one set of operations or another based on a condition. Being able to represent a problem-solving process as an algorithm is becoming increasingly important in any field that uses computing as

a primary tool (business, economics, statistics, medicine, engineering, etc.). Success in these fields requires algorithm design skills.

- As computers become essential in a particular field, more domain-specific data is collected, analyzed and used to make decisions. Students need to understand how to find the data; how to collect it appropriately and with respect to privacy considerations; how much data is needed for a particular problem; how to remove noise from data; what techniques are most appropriate for analysis; how to use an analysis to make a decision; etc. Such data skills are already required in many fields.

These computational thinking skills are becoming more important as computers, algorithms and data become ubiquitous. Coding will also become more common, particularly with the growth in the use of visual programming languages, like Blockly, that remove the need to learn programming language syntax, and via custom blocks, can be used as an abstraction for many different applications.

One way to represent these different skill sets and the students who need them is as follows:



All students need digital literacy, many need computational thinking depending on their career choice, and some will actually do the software development in high-tech companies, IT departments, or other specialized areas. I don't believe all kids should learn to code seriously, but all kids should try it via programs like code.org, CS First or Khan Academy. This gives students a good introduction to computational thinking and coding, and provides them with a basis for making an informed decision on whether CS or IT is something they wish to pursue as a career.

www.TechsCool.org #LearnToCode; Code To Learn