

Will Gillette

Dr. Mongan

CS-471: CS Pedagogy

2 November 2023

Cryptography: Incorporating Universal Design for Learning and CS4All Principles

Creating an inclusive computer science environment has been a struggle since the incorporation of computer science education in the classroom. Initially learning computer science comes with many unique obstacles such as working through syntax errors, understanding how to operate a computer effectively, and developing the problem-solving skills to carefully design algorithms to accomplish tasks. Furthermore, those unfamiliar with computer science can struggle with impostor syndrome when placed in the same classroom as those who built computers during their childhood. The principles of Universal Design for Learning (UDL) aim to remedy these issues by serving as a guide for instructors to cultivate inclusive learning environments for a variety of disciplines. Computer Science for All (CS4All) is an initiative with the mission of applying these principles to computer science education programs, promoting an equitable computer science learning experience for all students. While learning computer science initially comes with a variety of challenges, integrating these principles into every computer science curriculum will ultimately encourage more students to continue studying computer science throughout their lives.

The goal for UDL is to essentially promote inclusivity through appealing to people of all knowledge and ability levels, making lessons accessible regardless of them (“UDL Guidelines”). The first aspect of UDL discusses the significance of engaging all classmates through selecting

activities that are age appropriate, allowing them to set academic goals, fostering a collaborative environment where people value everyone's skills and ideas, and incorporating multiple means of feedback through formative assessments or self-assessment. The second umbrella term is "representation," referring to providing several different explanations and examples of concepts to appeal to diverse types of learners, emphasizing syntax, and encouraging "big picture" analyses. The next category discusses action and expression, which dives into the importance of varying the types of activities to appeal to students. For example, in one lesson, an instructor could incorporate kinesthetic learning, while in other, they could utilize plugged activities to appeal to tech savvy students. Furthermore, it is important to keep these plugged activities accessible to all students, ensuring that there are assistive technologies available to guide them. It is the responsibility of computer science educators to incorporate these ideas into their curriculum to encourage students to pursue further computer science studies.

CS4All makes it their mission to apply these principles to make computer science accessible to all students, including those with disabilities or who are English language learners ("How Might I Proactively Differentiate My CS Instruction to Reach All Learners?"). To achieve these goals, CS4All first follows the Web Accessibility Initiative, providing multiple means of accessibility features for plugged activities, as discussed in the final category of UDL. To create a meaningful learning experience, CS instructors must additionally become aware of the prior knowledge of all students, allowing them to engage each respective student by creating activities that match their prior experience. Pre-assessments are sufficient tools to gather this information, providing instructors with accurate quantitative data regarding their students' current abilities. With the variety of computer science backgrounds, grouping students together through pair programming or assigning roles based on strengths can be beneficial to their

learning. Pair programming enables students with similar skill levels to collaborate and learn together, making students of all learning backgrounds feel a sense of belonging and resulting in higher quality code (Cockburn). Finally, it is important to not only formatively assess students to ensure that they are effectively following along with the material but also to celebrate their completed projects and work, giving them opportunities to share them with their classmates, ultimately making them feel valued as a student even if they are not the most knowledgeable in the room. Through promoting the principles of utilizing both unplugged and plugged activities, checking in with students periodically, ensuring that there is assistive technology, and grouping students of similar skill levels, CS4All ultimately aims to create an inclusive environment that encourages students to continue with their studies of computer science.

When crafting a computer science lesson plan, instructors need to be mindful of both the goals of CS4All and the UDL principles to appeal to the largest proportion of their students. Project TACTIC describes the importance of integrating UDL principles progressively, starting with a single lesson or unit to determine which activities are the most successful in contributing to students' learning ("Project TACTIC: Teaching All Computational Thinking through Inclusion and Collaboration"). To elaborate, in CS education, instructors can provide multiple means of representation through both video examples that describe concepts directly and unplugged activities that can, for example, demonstrate algorithms. This ties into CS4ALL's goal of creating meaningful computer science learning experiences for everyone by providing multiple means of representation. One challenge of initially learning computer science is that students will tend to run into syntax errors that can create frustration, causing them to feel inadequate. Through both tools such as Code Assist and pair programming, however, students can resolve problems more efficiently, allowing them to continue progressing through their tasks

(“Achieving a Differentiated Computer Science Classroom: An Approach to Promote Equity”).

Hence, this assistive technology proves useful in helping students without prior knowledge learn the fundamentals of coding, aligning with both UDL principles and the goals of CS4All. Finally, as stated previously, by developing lesson plans that adhere to the mission of CS4All, instructors are automatically incorporating principles of UDL. Thus, there is a strong relationship between both the principles of UDL and the mission of CS4All, and instructors need to take advantage of these frameworks to appeal to a variety of students.

To apply these principles to my group’s lesson about cryptography, we aim to communicate the learning goals concisely, conduct both plugged and unplugged activities, and promote reflection and inquiry for students to understand the major takeaways from the lesson. Some of our essential questions include: “What is cryptography, and how does society use it?”, “How can individuals examine character frequencies as a tool for decryption?”, and “How can businesses use cryptography to secure classified information?” Following our presentation of these questions, we will showcase the code.org Vigenère Cipher widget, appealing to tech savvy students who do not have much trouble learning concepts presented digitally. An unplugged activity demonstrating how to encode and decode messages through using a handheld clock-based tool will supplement this, targeting the remainder of the students who struggled with following the widget. We will then dive into the significance of frequencies, using the code.org Frequency Analysis Widget to demonstrate the association between character frequencies in the English Alphabet and those of an encrypted message. During this aspect of the lesson, we plan to have students “guess-and-check,” with the goal of them identifying this association. Finally, we plan to conduct an engaging unplugged activity, involving grouping students and having them collaborate to crack a code to open a locked box, which will require them to apply decryption

methods. Finally, we will have students reflect on what was easy and difficult and how these concepts can apply to the real world. Thus, this lesson will target various aspects of both UDL and CS4All such as the importance of collaboration and providing multiple means of representation.

There are many natural challenges that come with creating a computer science curriculum that helps everyone understand the underlying concepts, but through incorporating the principles of UDL and the ideas from CS4All, computer science instructors can work to create an equitable learning environment for students of all knowledge levels. Through conducting both plugged and unplugged activities, celebrating students' accomplishments, and encouraging collaboration through either pair programming or assigning individual roles, students will gain self-confidence and shift their mindset about computer science. As a result, more students will want to pursue higher education in computer science, promoting the overall growth of this area of academic interest.

Works Cited

“Achieving a Differentiated Computer Science Classroom: An Approach to Promote Equity.”

TechSmart, <https://www.techsmart.codes/achieving-a-differentiated-computer-science-classroom-an-approach-to-promote-equity/>. Accessed 2 Nov. 2023.

CAST. “UDL Guidelines - Version 2.2.” Accessed 11/2/2023, <https://udlguidelines.cast.org/>.

CS4All Blueprint. “How Might I Proactively Differentiate My CS Instruction to Reach All Learners?” Accessed 11/2/2023, <https://www.csforall.org/blueprint>.

Cockburn, Alistair. “The Costs and Benefits of Pair Programming.” North Carolina State University, [Collaboration.csc.ncsu.edu](https://collaboration.csc.ncsu.edu/laurie/Papers/XPSardinia.PDF), <https://collaboration.csc.ncsu.edu/laurie/Papers/XPSardinia.PDF>. Accessed 2 Nov. 2023.

“Project TACTIC: Teaching All Computational Thinking through Inclusion and Collaboration.”

CTRL Education - University of Florida, <https://ctrl.education.ufl.edu/wp-content/uploads/sites/5/2020/06/CTRL-TACTIC-UniversalDesign.pdf>. Accessed 2 Nov. 2023.