

Planning Computing Instruction through the Universal Design for Learning (UDL) Framework

Maya Israel, Ph.D., University of Illinois at Urbana Champaign, misrael@illinois.edu
 Quentin Wherfel, M.Ed. University of Illinois at Urbana Champaign, wherfel2@illinois.edu
 Saad Shehab, M.Ed. University of Illinois at Urbana Champaign, shehab2@illinois.edu

The table below provides examples of how the UDL framework can be used by teachers to proactively plan for the learner diversity in their classrooms. It is adapted from the CAST (www.cast.org) UDL framework.

Multiple Means of Representation	Multiple Means of Action and Expression	Multiple Means of Engagement
<p><i>1. Provide options for perception</i></p> <ul style="list-style-type: none"> -Model computing lessons using an interactive whiteboard, videos, or already created templates -Give access to modeled code while students work independently -Provide access to video tutorials of computing tasks 	<p><i>4. Provide options for physical action</i></p> <ul style="list-style-type: none"> -Provide teacher’s codes or use partially completed code as templates -Include unplugged activities that teach through physical representations of abstract computing concepts -Use assistive technology including larger/smaller mice, touch-screen computers, screen readers 	<p><i>7. Provide options for recruiting interest</i></p> <ul style="list-style-type: none"> -Give students choices (project, topic, display of project). -Allow students to make projects relevant to culture and age -Minimize common “pitfalls” for both computing and integrated content by considering barriers to learning from the beginning of the planning process.
<p><i>2. Provide options for language mathematical expressions, and symbols</i></p> <ul style="list-style-type: none"> -Teach and review content specific vocabulary if computing is integrated into other content areas. -Teach and review computing vocabulary (e.g., algorithm, decomposition, abstraction) 	<p><i>5. Provide options for expression and communication</i></p> <ul style="list-style-type: none"> -Give options of computing software based on accessibility and engagement (e.g., Block based languages like Scratch, Etoys, Code.org, Alice vs. text-based languages) -Give opportunities to practice computing skills and content through projects that build on prior lessons 	<p><i>8. Provide options for sustaining effort and persistence</i></p> <ul style="list-style-type: none"> -Remind students of both computing and content goals -Provide extensions for students to keep engaged. -Encourage and teach peer collaboration (e.g., pair programming)

<p><i>3. Provide options for comprehension</i></p> <ul style="list-style-type: none"> -Activate or build background knowledge by making computing tasks interesting and culturally relevant -Clearly state lesson content and computing goals -Involve students during teacher modeling by asking questions as comprehension checkpoints 	<p><i>6. Provide options for executive functions</i></p> <ul style="list-style-type: none"> -Guide students to set goals to accomplish when they have a project that will require multiple days or weeks -Develop methods to record each student's progress (have planned checkpoints during lessons for understanding and progress for computing skills and content) 	<p><i>9. Provide options for self-regulation</i></p> <ul style="list-style-type: none"> -Communicate clear expectations in terms of what students will work on, how they will work (individually or with peers), how they can seek help -Develop ways for students to self-assess and reflect on own projects and those of others
---	---	---

Adapted from www.cast.org, www.udlcenter.org. Copyright 2011 by CAST

***There are many resources about using the UDL framework during instruction. This work can be applied to CS education. The table above is a working document that is dynamic and can change. If you have suggestions, examples, developed lesson plans using UDL during CS instruction, etc., please let me know so we can start building this body of practitioner work.

Thank you,
 Maya Israel
 University of Illinois at Urbana Champaign
misrael@illinois.edu

