DIFFERENTIATING FOR DIVERSITY:
USING UNIVERSAL DESIGN FOR LEARNING IN COMPUTER SCIENCE EDUCATION

Alexandria K. Hansen, UC Santa Barbara
Eric R. Hansen, SPED Educator
Hilary A. Dwyer, University of Colorado Boulder
Danielle B. Harlow, UC Santa Barbara
Diana Franklin, University of Chicago

What does FOR ALL really mean?
How do we actively support students who do not already have opportunities to learn computer science.
Populations left out:
- Schools without computer labs or devices
- No one is qualified to teach CS
- Students unable to use mouse or keyboard
- Visual impairments
- Hearing impairments
- ADHD
- Autistic students

This includes, but is not limited to:
- Students learning English
- Students from underrepresented minority groups
- Students with learning differences (and/or diagnosed learning disabilities)
- Students below grade level
- Students raised in low-income families
- Without regular access to technology
- With low motivation or perseverance

What happens when you design for equity?

“Flexible approach” to teaching in which the teacher plans and carries out varied approaches to content, process, and product in anticipation of and in response to student differences in readiness, interests, and learning needs” (Tomlinson 2001)

Highlights the need to design aspects of instruction that are:
- Necessary for SOME,
- Beneficial for ALL, and
- Not detrimental to ANY.
Ways to Differentiate for ALL Students

1. Universal Design for Learning (UDL)

Highlights the need to design aspects of instruction that are:
- Necessary for SOME,
- Beneficial for ALL, and
- Not detrimental to ANY.

KELP-CS
Kids Engaged in Learning Programming & Computer Science

Purpose
1. Teach computational thinking, programming, and engineering design in upper elementary classes during the academic day.
   - Programming Environment: La Playa
   - Module 1: Digital Storytelling
   - Module 2: Game Design
2. Develop and test learning progressions about how students engage with and learn computer science.
   - Algorithms, knowledge of computers, programming, thinking about the user, data and abstraction.

The Curriculum (Grades 4 – 6)

The Curriculum (Grades 4 – 6)
Designing for English Proficiency

- Simplified & clarified instructions.
- Reduced required reading/writing on student worksheets.
- Embedded more text within the interface.
- Added audio read-aloud function

For more information, see:

Designing for Math Proficiency

- Removed negative numbers, decimals and percentages.
- Changed location of origin within coordinate plane system and added a grid function.

For more information, see:

Designing for Varied Student Pace

Created the Sandbox – a designated, open-ended play area to experiment and practice for students who finished assigned tasks early.

Designing for Culture

- Conducted interviews with students to ensure examples were culturally-relevant (e.g., piñatas, quesadillas).
- Ensured programmable characters of every ethnicity were available.
- Changed language of the “tortoise and hare” project, and included the story.

Accommodations

Re-teaching & Small Group Instruction

- When many students were confused, re-teaching the class in a different way was necessary.
- When fewer students were confused, small group instruction was used.

Accommodations

Modeling

- In lessons that proved difficult for many students, instructors often acted out programming commands.
- Modeling can also refer to the process of showing students how to program something, explicitly modeling the thinking behind each action.

1/28/2014: Analytical Memo
"There was still a lot of confusion over how to use...the glide blocks. I tried to mimic [the activity] to help. They had to "program" me to walk to the nearest bookshelf. Unless they said glide, I didn’t actually move forward."
Modifications For Struggling Students

- The Sandbox was also a great differentiation tool for struggling students who needed a break.
- It also allowed teachers to create small, individualized assignments that better aligned to a student’s current skill level.

4/30/2015: Teacher Interview

“The Sandbox was great because I could direct my struggling students there. If a student had a difficult time completing the lesson, they could take a break in the Sandbox and play. Or... I could create smaller, personalized assignments in the Sandbox. Instead of making the car go up, down, right, and left with arrow keys, maybe we only try to get the car to move right.”

Modifications For Advanced Students

- Some students were identified as computer helpers.
- These students finished work early and enjoyed helping others.
- They were free to walk around the classroom, helping peers.

Hearing impaired:

- https://www.hearinglikeme.com/hearing-loss-simulator/

Hearing impaired – UDL Suggestions

- Don’t depend only on music for mood – change coloring or provide other visual cues as context
- Provide text for speech that occurs in game

Sight impaired:

- https://simulator.seenow.org/
<table>
<thead>
<tr>
<th><strong>Sight Impaired – UDL Suggestions</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Make sure colors are visible for different types of color blindness</td>
</tr>
<tr>
<td>• Make icons large with bold lines</td>
</tr>
<tr>
<td>• Don’t clutter the interface</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>English Language Learners:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Provide visual cues for actions</td>
</tr>
<tr>
<td>• Tutorial walk-through rather than direction</td>
</tr>
<tr>
<td>• Text-to-speech capabilities</td>
</tr>
<tr>
<td>• Definition functionality (or example)</td>
</tr>
<tr>
<td>• Using purely visual ways of conveying information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Attention Deficits:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Don’t have long blocks of text w’out images</td>
</tr>
<tr>
<td>• Have the option to replay things in case they miss them</td>
</tr>
<tr>
<td>• Forms / submitting text – say how long they will take and have option to save progress</td>
</tr>
<tr>
<td>• Require user input to advance so user is ready</td>
</tr>
<tr>
<td>• Break up game into small chunks / levels</td>
</tr>
<tr>
<td>• Simplify user interface</td>
</tr>
<tr>
<td>• Don’t make time limits too important</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Cognitive Impairment:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Split into steps and provide instructions for different steps</td>
</tr>
<tr>
<td>• Allow access to previous levels (with instruction)</td>
</tr>
<tr>
<td>• Checklists</td>
</tr>
<tr>
<td>• Hint button</td>
</tr>
<tr>
<td>• Organize it well so they aren’t overwhelmed by information</td>
</tr>
</tbody>
</table>
Get into project pairs

- Brainstorm ways that you either have or could integrate accommodations into your design.
- Make a list, then share out.

http://www.color-blindness.com/coblis-color-blindness-simulator/

This includes, but is not limited to:
- Students learning English
- Students with learning differences (and/or diagnosed learning disabilities)
- Students below grade level
- With low motivation or perseverance
- Cognitive impairments
- Visual impairments
- Auditory impairments