

Best Practices for Teachers using CS Platforms

This list contains access considerations for students which are specific to computer science applications. Be sure any web-based application used in an educational setting complies with the [Web Content Accessibility Guidelines \(WCAG\)](#), which have been developed by the World Wide Web Consortium (W3C).

JUMP TO

[Universal Access](#)

[For a student who is deaf or hard of hearing](#)

[For a student who is blind or visually impaired](#)

Universal Access

	Best Practices
Pedagogy	Multiple options for scaffolding and assessment are provided, such as unplugged activities, coding challenges that are chunked as pieces of an entire project, etc.
User Interface	Consistent, simple layouts allow easy access to resources for locating and understanding proper code syntax. All user interactions have meaningful keyboard shortcuts, and are grouped into logical regions.
User Interface	Visual and auditory feedback is provided to confirm when copy, paste, cut, or delete is performed.
Program Input	Alternatives for input and editing include mouse, keyboard, switches, eye gaze, touch, voice, and other assistive technologies.
Program Input	Methods for assembling code have tactile options in addition to text, such as blocks that link together or print outs of coding blocks
Program Output	Output is auditory, visual, and kinesthetic in addition to or instead of being on screen only. <i>Suggestion: Add a robot that lights up, makes noise and moves in addition to using on screen only code.</i>
Program Output	All output, including the results of prior entries, should be navigable using the keyboard, and vocalized appropriately.

For a student who is deaf or hard of hearing (DHH)

	Best Practices
Spoken instructions, descriptions, etc.	Instruction and all audio content are captioned and provided with ASL interpretation. CS terminology is translated clearly and consistently into ASL. Provide a glossary of terms, including both written definitions and ASL signs. <i>If no ASL translation is available, check captions to make sure they are reading level appropriate. If not, provide an appropriate transcript.</i>
User Interface	All auditory cues (beeps, bonks, etc) should have a visual cue as well.
Program Output	When the output of code contains an auditory component (such as a buzzer), a visual element (such as light) is available as well.

Two websites for computer science sign language:

[100BaseT – STEM Dictionary](#) from RIT
[Entries | ASLCORE Computer Science](#)

For a student who is blind or visually impaired (BVI)

	Best Practices
User Interface	All menus, buttons, dialog boxes, regions, etc a programming environment (output, input, coding syntax library) are navigable in a logical manner using a screen reader and/or keyboard shortcuts.
User Interface	Messages for program output and coding errors are auditory as well as visual (icons, color changes).
Program Input	Switching from navigation mode to programming mode is straightforward when using a screen reader
Program Input	Code is editable using only a screen reader and the keyboard and/or a braille input device
Program Input	Syntax can be navigated letter by letter and clearly voices special characters (like brackets), changes in letter case, color that is significant and other important syntax
Program Input	Students can drop “breadcrumbs” when navigating programs. These breadcrumbs can be navigated and removed via keyboard shortcuts.
Program Input	Line numbers and syntax highlighting can be vocalized via keyboard shortcut
Program Input	Code is vocalized using meaningful, age-appropriate text, rather than forcing students to listen to - and parse! - raw syntax
Program Input	Drag and drop coding can be done with or without using a mouse.
Program Output	Program output is vocalized in meaningful ways: • If the output is textual, it should be vocalized <i>along with the datatype</i>. For example, “3, a Number” vs. “three, a String”. • If the output is graphical, a best-effort is made to describe it. • When possible, programmatically generated animations should describe only what changes from frame to frame
Program Output	When the output of code contains a visual component (such as a light), an auditory element (such as a beep) is available as well.
Program Output	A screen reader voices the grid coordinates when there is any output in a grid form (such as a maze or a cell phone screen for coding an application.) A file is provided to create a tactile reproduction of on screen grids or environments.

Other resources to consider:

[No mouse challenge](#)

[Accessibility posters](#)