Adolescent Identity Expression through Transdisciplinary, Computational, Creative Writing

Adrienne Gifford
Open Window School
Bellevue, WA United States
adrienneg@ows.org

Sophia Dahl
University of Washington
Seattle United States
sdahl4@uw.edu

Cara Pangelinan
University of Washington
Seattle United States
cararene@uw.edu

Amy J. Ko
University of Washington
Seattle United States
ajko@uw.edu

Abstract

Background. Prior work has often speculated about learning synergies between computer science and language arts, building upon parallels between natural languages and programming languages and between writing and programming. Studies of primary classrooms have found applications of CS learning to reading literacy and applications of programming to interactive storytelling.

Objectives. While prior work has richly explored youth learning, it has not explored opportunities in adolescent spaces. We sought to understand what additional synergies might be possible in relation to student culture and identity, particularly when CS and language arts are combined in a transdisciplinary way by leveraging programmable media that specifically centers text, language, culture, and writing. Our study examined what learning can occur and what structural factors in teaching, pedagogy, and programming media influenced this learning.

Methods. We taught a 6-week, 20-hour transdisciplinary creative coding and writing course with seven 14-16 year old students using Wordplay, an educational programming language for creating multilingual interactive typography expressions. We captured student work samples, student reflections, and teacher reflections, and conducted an inductive thematic analysis on what tensions and opportunities arose between the two disciplines.

Results. We found that students employed code to express their identity in ways that text alone could not capture, that disciplinary writing practices around outlining and planning mutually reinforced cross-disciplinary learning, that the richer identity expression encouraged peer learning, and that integration enabled writing-interested students to develop interests in CS, and CS-interested students to develop interest in writing.

Conclusion. These findings suggest that creative writing and CS are synergistic not only for children, but for adolescents as well, particularly when programmable media centers on textual, typographic, and written form.

ICER 2026 Vol. 1, Uppsala, Sweden

ACM ISBN 979-8-4007-2203-5/2026/08
https://doi.org/10.1145/3765964.3811642

CCS Concepts

• **Social and professional topics** → **K-12 education**; **Computing literacy**.

Keywords

interdisciplinary learning, creative coding, project-based learning

ACM Reference Format:
Adrienne Gifford, Sophia Dahl, Cara Pangelinan, and Amy J. Ko. 2026. Adolescent Identity Expression through Transdisciplinary, Computational, Creative Writing. In *Proceedings of the ACM Conference on International Computing Education Research Vol.1 (ICER 2026 Vol. 1), August 11–14, 2026, Uppsala, Sweden.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3765964.3811642

1 Introduction

In past decades, there has been a rise in explorations of curricular integrations between CS and other disciplines. This is partly due to disciplines themselves changing to engage CS concepts (e.g., natural and social sciences embracing data science methods), but also due to intentional efforts to explore these disciplinary shifts through integrated curriculum. For example, 'CS + X' curricula in post-secondary settings have broadly sought to contextualize computing in the context of other disciplines such as mathematics, natural sciences, social sciences, and the arts [6, 7, 25]. Parallel efforts in K-12 such have deeply explored curricular, teaching, and learning synergies between CS and mathematics [52] through curricula like Bootstrap Algebra, and CS and social sciences, with critical accounts of CS that reframe foundational CS concepts through a social lens [35]. These explorations reflect the much longer history of educators and education researchers who have extensively studied the integration of multiple disciplines in the K-12 classroom [16, 27, 57], finding growing evidence that interdisciplinary teaching can improve student learning across disciplines [11, 24, 28, 49].

Most of these prior works on integration have focused on CS integrations with science, engineering, mathematics, and the arts [48]. However, an increasing body of work situated in primary and early secondary education has explored integrations between CS and the Humanities, with a wide range of learning objectives. Some works have positioned Humanities as a way to deepen ethical reasoning and cultural understanding about technology [14]. Others have positioned Language Arts (LA) as a promising context for introducing foundational CS concepts, arguing that narrative, syntax, and abstraction in writing parallel key ideas in programming [44].

Approaches to CS and LA integration vary in their goals, but often focus on forms of creative expression. Studies of *Storytelling Alice* explored how narrative based programming could utilize creative writing practices as a way to sustain student motivation [32]. Other works have examined this relationship in the reverse direction, investigating how iterative and procedural programming practices support students' LA learning [9]. Other works have identified concrete ways that Scratch can be a media context for introducing basic LA concepts of narrative and conflict to primary school youth [46, 54]. Some have even positioned programming as a kind of language, to augment young children's language learning in general [5]. These integration approaches have evolved into pedagogies that intricately weave together CS and LA in ways that can support learning in both disciplines simultaneously, by focusing on the complementary elements of writing and interactive stories [31].

While this prior work with young learners' creative storytelling with computation is rich and promising, there has been little research examining integrations at the secondary level, with adolescents. This is partly structural, as most LA and CS classes in secondary exist under curricular and standards expectations that constrain integration more than at the primary level, where integration is the norm. But this gap is a potentially missed opportunity to understand what other synergies might exist at adolescent ages, where students are more actively engaged in identity development and expression, particularly in the context of school [61]. For example, the stories young children might want to express in elementary school might be different than those they would tell at age 14-17, as might their willingness to tell those stories. Similarly, the narratives that adolescents tell about their identity in relation to CS and computer programming might be disrupted, if the telling of those stories was aided by computing and its concepts, rather than distant.

To understand the potential for transdisciplinary LA and CS integration at these ages, we conducted a design-based research inquiry [8] into two questions:

(1) What learning happens when we combine LA and CS learning in a transdisciplinary, secondary learning environment?
(2) What structural factors in teaching, pedagogy, and programmable media influence those experiences?

To answer these questions, we designed a 6-week, 20 hour summer elective course for 7 adolescents aged 14-16 that co-taught writing and programming principles. The course used a programming language designed for LA and CS integration called Wordplay, which uses code to make text and typography animated and interactive, centering language and writing as the core media. The course was designed with the underlying premise that teaching the two disciplines together could enhance students' ability to tell their stories in new and different ways, while also promoting CS learning. Overall, we found that it did, but that the transdisciplinary nature of the learning also deepened peer learning and cross-disciplinary interests.

2 Background

Below we review the multiple discourses on inter- and transdisciplinarity, and prior work on integration of computing with other disciplines.

2.1 Transdisciplinarity

Disciplines can bring with them disciplinary blinders: knowledge and values that reflect specialized training, more than universally applicable truths. For example, some have argued that modern challenges, and therefore modern education, require more multi-faceted problem solving skills [43]. Educating students to be prepared for this reality is something that has been recognized by educators for some time, fostering a long history of advocacy around transdisciplinarity [3]

The practice of disciplines being studied in isolation has created norms for when to enact certain disciplinary practices, often leading to patterns of subjugation of knowledge to certain learner demographics [23]. For example, students who had a background in using *meter* in poetry would not have the opportunity to utilize that understanding of language structure in coding language syntax (e.g. indentation-sensitivity). To explore the opportunities of learning without disciplinary boundaries, researchers have called for new approaches to disciplinary learning that promote the integration of multiple disciplines [28].

We draw on Nicolesiu's transdisciplinary philosophy [42], which prioritizes the unity of knowledge beyond disciplines as a way to address the problem of fragmentation of knowledge, bridging the sciences, engineering, and humanities disciplines. We also draw upon the education framework created by the Transdisciplinary School within the University of Technology Sydney (UTS) [34], which focuses learning on eight key synthetic capabilities for students: 1) *Analytic* (understanding complex citations), 2) *Creative* (imagining and exploring alternatives), 3) *Experimental* (testing ideas in real-world contexts), 4) *Normative* (exercising judgment to create value), 5) *Transdisciplinary know-how* (assembling a transdisciplinary process and justify choices), 6) *Relational* (Acting with contextual, inter subjective awareness and holistic understanding of complexity), 7) *Reflexive* (understanding self in relation to others and the world), and 8) *Agenetic* (acting purposefully). Unlike frameworks that measure based on outcomes and observable disciplinary behaviors (e.g., solve this algebra problem, write a coherent essay), this framework for assessing outcomes that are greater than the sum of their parts. We used this framework to design our classroom's culture, aims, and pedagogies, centering student growth and unification of disciplines over skill mastery.

2.2 Transdisciplinarity in STEAM

Much prior work on transdisciplinary integrations between CS and other disciplines has been through the arts, often described as STEAM (Science, Technology, Engineering, Arts, and Math) contexts. Many prior works have found evidence of learning across these five disciplines, while fostering creativity [1, 4, 41, 55, 56]. For example, in their paper on transdisciplinary teaching, Finch et al. [20] observe how new genres and tools, such as e-textiles and data sculptures, create opportunities for transdisciplinary learning. But the authors also note the risks of "asymmetric valuing" of art and STEM, where art is often used in service of STEM competency.

2.3 Technology in LA

In LA education research, there has been substantial discourse on teaching "new literacies" to students. "New literacies" are defined as literacies that emerge due to how literacies are embedded in changing social practices, cultural contexts and identities [40]. Over time, this has come to include "digital literacies," which emphasize the essential skills for managing and communicating information in a rapidly developing digital world [58]. Several prior works have shown that integrating digital literacies into LA learning has acted as a motivator for students to attempt new paths of learning, stay engaged with class material, as well as learn relevant LA skills [29, 40, 50]. In this way, digital technologies have been asymmetrically used in service of LA competencies.

2.4 LA and CS Integration

Several have argued for transdisciplinary forms of LA and CS integration. For example, Lynch argued that CS *is* a language art, and for a reconceptualization of the discipline in LA terms [38]. This argument advocated for a "creativity first, programming second" style of classroom learning, describing 'BardBots', where youth were asked to read and interpret Shakespeare plays, and then program robots to perform the scene. The creative aspects with the inclusion of LA helped students become more confident, reflective and articulate about both programming and plays.

In a similar way, a diversity of prior works have explored using CS as a creative tool for LA expression [9, 32, 46, 54], and LA skills as a foundation for CS learning [5, 31, 44]. Other work has studied CS with LA in undergraduate curricula, noting integration threats, where teachers often fear that one discipline's learning goals are compromised to make space for CS learning goals [36]. This work and others, however, have found that when CS is used as a tool (e.g., using code through a digital humanities approach to understand texts), students still met learning goals for learning areas while learning CS skills as well [10].

Another way conceptualize LA integration into CS is as a culturally responsive pedagogical strategy [21]. In this framing, LA is engaged in service of CS learning, to explicitly address the many ways in which de-contextualized, non-responsive CS curricula embed cultural biases that speak only to a specific subset of youth [39]. When engaged in introductory contexts by engaging culture, identity, and systems, such pedagogies can not only better engage a diversity of youth, but also produce robust learning of CS concepts and skills that even exceed the narrow learning goals of programming-focused introductory courses [53]. Prior work in CS education has engaged many disciplines in responsive ways, including fashion through fractals [18], craft through geometry [19], social sciences through everyday life [51], and health and nutrition through machine learning [60]. This paper explores such pedagogy by connecting CS to youth's identity work through creative writing.

2.5 Creative Writing in LA

Creative writing pedagogy is still a relatively new field compared to the study of English literature as a discipline. In the U.S., creative writing was first taught at a junior high school in the early 20th century as an alternative to teaching literature and was designed to appeal to the youth of that time. In the 1930s, the University of Iowa became the first U.S. higher education institution to offer creative writing as its own discipline, and it is now commonly taught as a standalone option to pursue (e.g., through Master of Fine Arts programs).

Education discourse on creative writing in LA prioritizes pedagogy that centers students' critical thinking and communication skills, since so few students go on to be published authors [15]. The majority of prior works describe observations and teaching experiences, but not student experiences or expressions [12, 22, 29, 30, 33]. Only one prior work described how their pedagogy benefited students [29], relying on interviews and observations from educators to detail how digital storytelling provided students with new and different opportunities for self-expression and agency. Other prior work has critiqued creative writing workshops [37] for putting too much emphasis on a writer's product as opposed to the writing process.

Some prior research has focused on revision and feedback in creative writing, which are relevant to the pedagogies used in our study. Zhang et al. [63], drawing upon 30 years of education research on writing, discuss the relationship between student writing goals to revision with self regulated revision, peer revision, and self reflection obtained from reviewing others' writing in the context of early undergraduate education. They found that high-level revisions are commonly associated with writing goal completion, and peer comments made the largest singular contribution to achieving revision goals. Similarly, Wooley et al. [62], measured the effects of elaboration and prototypical examples during the reviewing stages of writing. They claim that students who provided elaborate forms of feedback which included free form comments performed significantly better on their own writing than those who provided numerical ratings only. In their work, peer revision created learning benefits for all involved by promoting a deeper understanding of writing structure, and fostering critical analysis about communication.

More broadly, in the U.S., creative LA has struggled to have a presence in education. The introduction of *Common Core* standards, for example, leaves little room for creative writing within the middle and high school English classroom [17], calling into question how to make space for creative writing in school settings that emphasize career and college readiness over creative endeavors.

3 Methods

This design-based research [8] study builds upon these many disparate prior works, framed around transdisciplinarity in its goals, building upon creative writing and CS integrations in primary, and conscious of the risks of asymmetry between CS and LA. Our goal in designing learning was for students to experience creative writing and computer science as *deeply connected practices* rather than as two siloed subject areas that happened to support each other. Drawing on transdisciplinary perspectives [34, 42], we treated creative writing and programming as complementary practices that students could use to express their identities, with the goal of demonstrating *Transdisciplinary know-how* through *Relational* and *Reflexive* awareness. Our exploratory research questions focused on what kinds of learning this framing would foster and how the learning design choices we made would structure this learning.

Our university's Institutional Review Board reviewed the project and determined that it qualified for exempt status on the condition of student anonymity and assent.

3.1 Instructional Design Process

We used a qualitative, design-based research (DBR) approach to explore the tensions and opportunities of integrating LA and CS. DBR is a methodology that studies learning in real-world educational contexts through cycles of designing, testing, and refining classroom activities in collaboration with teachers and students [2]. It is both practical and theory-informed, and required ongoing cycles of designing, testing, and adjusting activities as we built knowledge to improve our teaching practice.

Our study was based on a 6-week course structure. Co-design was central to our approach to envisioning these 6-weeks, following the many prior works on culturally responsive and sustaining CS pedagogy [35], which center student agency. Through weekly surveys and ongoing in-class feedback, we invited students to guide what we focused on, how project work evolved, and what kinds of support they needed. As a teaching team, we met regularly to review student feedback, reflect on how activities were working, and plan adjustments for the following class sessions. These ongoing conversations shaped the pacing of the course, the projects we selected, and the kinds of programming constructs that we introduced. This approach helped us explore how identity-centered creative writing and interactive coding could support one another, especially as students drew on their cultural, linguistic, and personal backgrounds in their work. Our goal was to create a space where students could draw from both disciplines to tell their stories and build on the knowledge and strengths they brought into the room.

Building upon Kligyte and Nicolescu's framing of transdisciplinary learning as [34, 42] fundamentally synthetic, we selected learning activities that would give opportunities for integrating identity and reflection across disciplines. We adapted writing activities from *Reading, Writing, and Rising Up: Teaching About Social Justice and the Power of the Written Word* [13], which we selected for their focus on identity expression and building an inclusive classroom community. These assignments encouraged students to connect their classroom activities and projects to their own lives and identities. For example, students started the course by writing a "Name Poem" about the stories and meanings connected to their names.

Because our course centered on identity expression, it was important to use a platform capable of supporting students' full linguistic diversity and range of access needs. We chose the educational programming language Wordplay [1]. It was an ideal choice for exploring LA and CS integrations, as it was intentionally designed for writing short programs that implement interactive, multilingual media with words, emojis, writing, and typography, across all of the world's languages. Wordplay was also accessible to students with a diversity of disabilities. These features aligned closely with our goals for an identity-centered and identity-inclusive transdisciplinary creative writing and CS course, where the point of the platform was not to learn CS concepts, but to express identity through CS concepts.

[1]Name and citation anonymized for review

3.2 Setting and Participants

The course took place within *Upward Bound*, a federally-funded U.S. college readiness program that serves low-income and first-generation high school students. We offered the elective course as part of the summer session, hosted at the University of Washington.

Eleven high school students initially enrolled in the course. Four withdrew early due to other obligations or medical reasons, and seven students completed the course. Our student group represented a mix of gender and ethnic identities and first languages, and all had chosen our course as an elective. Several students mentioned at the start that they had little to no programming experience, while others were eager to try connecting coding with creative writing for the first time. To protect anonymity, we refer to students by participant IDs (e.g., s1, s2...). Final feedback survey responses were anonymous, so those quotes are reported without participant IDs.

We co-taught the course as a team of three instructors, and we also served as the research team. One instructor was a certified computer science teacher with many years of classroom experience, another was a recently certified LA teacher, and the third had been involved in the design of the Wordplay platform. We were aware that these different backgrounds influenced how we approached planning and teaching, and we were intentional about balancing expertise. For example, we rotated leadership across writing workshops, programming activities, and platform support, ensuring that no single background dominated instructional decisions. Our shared commitment to a student-centered, co-designed, identity-expressive course made it easier to align our decisions with student needs. All three instructors met regularly to reflect on student feedback and adjust the curriculum in ways that responded to students' needs and ideas.

To keep the reporting clear, we refer to the three instructors throughout the paper as Instructor A, Instructor B, and Instructor C. The three instructors are co-authors of this paper; the fourth author was not involved in teaching the course, but provided research design guidance. When we draw on instructor reflections or debrief conversations, we identify the source using these labels.

3.3 Curriculum

3.3.1 Course Goals. In our initial design process, we identified four main goals for the course design: (1) support students in expressing their identities through creative writing, (2) introduce foundational programming concepts to reinterpret writing as interactive multimodal projects in Wordplay, (3) foster an inclusive classroom culture of constructive peer feedback, and (4) invite students to guide the course through ongoing co-design. These goals shaped both our weekly activities and the adjustments we made in response to student feedback.

3.3.2 Course Overview.

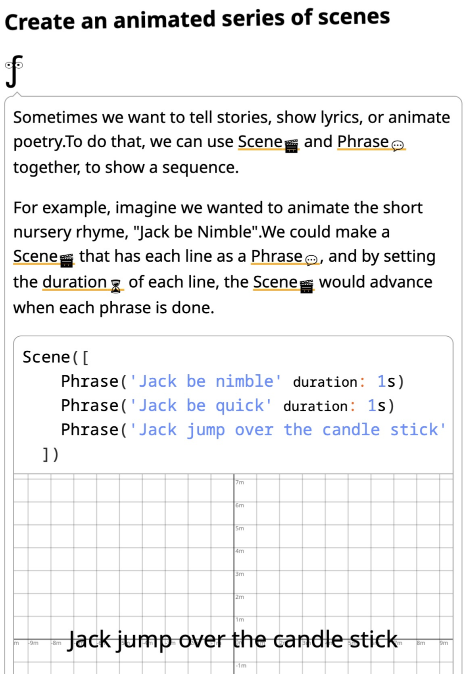


Figure 1: A starter template from Wordplay's "How-To" Guide, provided to introduce Wordplay's Scene construct, which displays a series of text phrases over time.

Week 1: Building Community and Identity Work. We focused on building community and learning about students' motivations, relationships with writing, and prior experiences with programming. Students began their first project, a written Name Poem drafted on paper or typed in a traditional writing format, drawing inspiration from provided writing samples and family conversations. Poems were shared aloud in a read-around, with peers offering written feedback. As each student read their work aloud, classmates wrote sticky notes with compliments or favorite lines.

Week 2: Translating Writing into Kinetic Poetry. Students began transforming their Name Poems into kinetic poetry. In this genre, readers interact with words as they animate and change on the screen. Students considered how they could use programming in Wordplay to animate their text with motion, timing, font, color and emojis to better express the meaning of their poems. We introduced Wordplay through a starter template (Figure 1) and short mini-lessons on functions and language translation, features in Wordplay's editor that allow students to add language-tagged alternatives to text, "skin" code in any language, switch between views of a program in different languages, and generate multilingual output. Students used graphic organizers (Figure 2) to identify the language translations, emojis, timing, sequencing, and movements they wanted to learn to program in Wordplay in order to express the mood, tone, and meaning of their poems. We also discussed emojis, examining whose cultures are represented in the Unicode Standard and how students could create their own characters using Wordplay's *Custom Characters* feature, which lets students design pixel and vector art emojis and symbols to use in their projects and share with others, supplementing Unicode.

Week 3: Studio Work and Debugging. Week three began with studio time to allow students to enhance their Wordplay projects by adding emojis, custom characters, movement, and other effects. Instructors introduced debugging strategies along with the core Wordplay functions students needed to create an animated poem. Because Wordplay is a functional programming language, we first reviewed the general concept of a function as well as how lists work in programming. We then introduced two foundational functions for programming an animated series of scenes in Wordplay: `Phrase`, an output function that creates text that can be styled or animated, and `Scene`, an input stream function that takes a list of outputs and shows them in sequence, over time. As students began coding their poems, we introduced additional Wordplay functions based on the effects they wanted to create, such as `Sequence` for keyframe style animations and `Group` for arranging multiple outputs on one screen. Later in the week, students engaged in a gallery walk. Students moved around the room to experience each other's projects and leave sticky notes of encouragement and questions. Students closed the week with reflections on both their creative writing and creative coding choices in this project.

Week 4: Final Project Writing Options. During week four, students focused on writing as they began their final project of the course. They chose between two options: a "Where I'm From" poem or a "Sweet Learning" narrative, both of which encouraged connections to home, family, identity, and personal experience. Most of the week was spent drafting their written pieces, receiving peer feedback, and revising their written work. At the end of the week, we introduced the concept of user interaction in Wordplay, explaining `Input` streams as the way a user or reader communicates with a program (e.g., pointers, keyboards, microphones). While students did not program during this week, we demonstrated the available input types in Wordplay and encouraged them to begin considering which inputs might best support their stories or poems when they

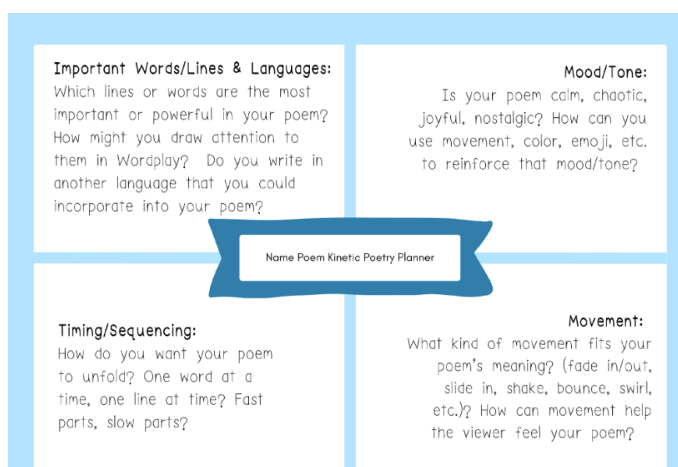
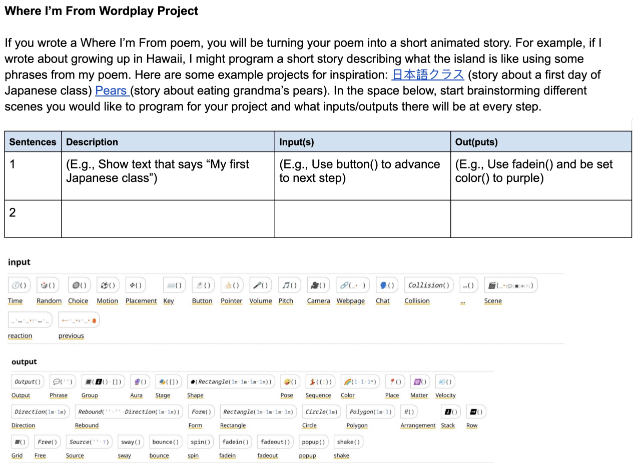


Figure 2: Graphic organizer used for planning kinetic poems.



Where I'm From Wordplay Project

If you wrote a Where I'm From poem, you will be turning your poem into a short animated story. For example, if I wrote about growing up in Hawaii, I might program a short story describing what the island is like using some phrases from my poem. Here are some example projects for inspiration: 日本語クラス (story about a first day of Japanese class) Pears (story about eating grandma's pears). In the space below, start brainstorming different scenes you would like to program for your project and what inputs/outputs there will be at every step.

Sentences	Description	Input(s)	Out(puts)
1	(E.g., Show text that says "My first Japanese class")	(E.g., Use button() to advance to next step)	(E.g., Use fadein() and be set color() to purple)
2			

Figure 3: The graphic organizer students used to plan interactive elements for for their final Wordplay projects.

develop their writing into an interactive Wordplay project in week five.

Week 5: Peer Review and Project Planning. Students revised their writing, exchanged peer feedback, and shared their written work in a second read-around during the first part of week five. Each student read their work aloud while classmates responded with "I like..." and "I wonder..." statements written on sticky notes. After this final writing phase, students completed a graphic organizer (Figure 3) to plan how their written ideas could be developed into interactive `Input` streams and `Output` values in Wordplay. Instructors used these organizers to determine which Wordplay Inputs to introduce in more detail, based on the inputs students indicated they wanted to use in their projects. This week's mini-lessons focused on the `Button` and `Key` inputs, which allow users to interact with the program through the primary mouse button, touch, or by pressing keys on the keyboard, and on the concept of conditionals, which naturally fit with students' use of `Button` and `Key` as they began designing interactions that depended on those inputs.

Week 6: Project Completion and Public Sharing. In the final week of the course students had extended studio time to complete their final Wordplay projects. During this time, instructors introduced two additional Wordplay functions, `Pointer`, which tracks the pointer position, and `Bind`, which lets students name values for reuse in their programs. Students also taught mini-lessons to their peers on techniques they had discovered on their own, including `Arrangement`, which controls how elements are positioned relative to one another in a `Group`, and an algorithm for creating rainbow color-changing text. The course concluded in a final Gallery Walk, where each student shared their interactive piece, exchanged peer feedback, and reflected on their learning. The Upward Bound program culminated in a banquet where students were proud to share their interactive poems and stories with their peers and family members. Figure 4 shows several students working on their final projects; Figure 5 shows a diversity of projects students worked on throughout the class.

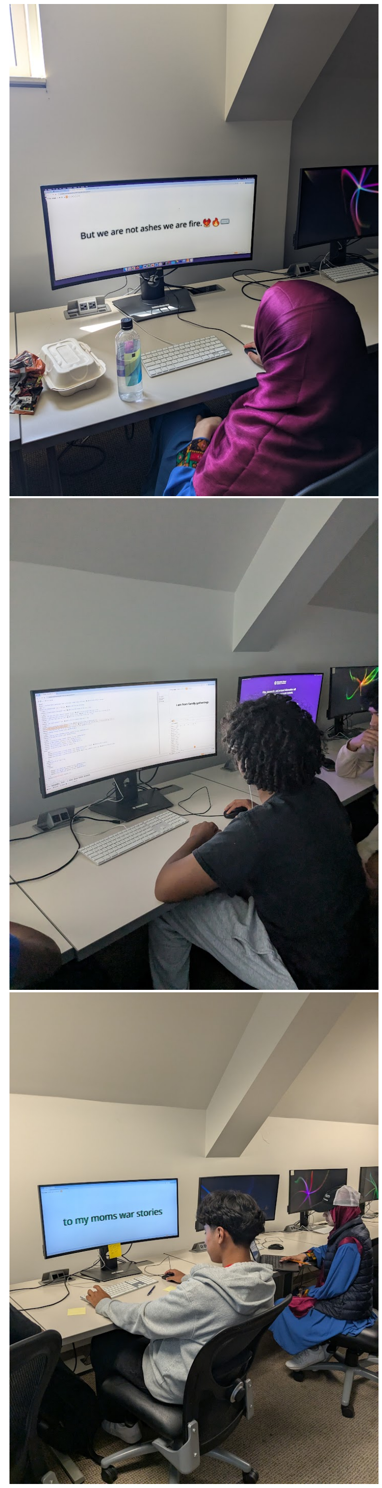


Figure 4: Students working on their final Wordplay projects during Weeks 5 and 6.

3.4 Data Sources

As Kligyte and Nicolescu conceptualize transdisciplinary learning as spanning creativity, analysis, relationality, and other synthetic skills [34, 42], we gathered reflective data that would allow us to observe this synthesis directly:

- *Student Work Samples.* Students completed two major writing assignments, two programming projects in Wordplay, and graphic organizers for planning. We archived all student work, allowing us to observe how students' creative writing and computational thinking developed over time.
- *Student Surveys.* At the end of each week, students completed surveys with open-ended questions about what worked well for them, what they found challenging, and what additional support they'd like.
- *Student Reflections.* After each major writing assignment or programming project, students wrote short reflections on their creative process, self-expression, and feelings about their work. At the end of the course, they also wrote longer reflections about their overall experience.
- *Teacher Reflections.* Instructors kept daily notes on collaborations, challenges, and opportunities in integrating language arts and computer science, student engagements, and any changes made to the curriculum or course design.

Because the teachers were also authors on this paper, we intentionally treated daily reflection logs as a form of data, using them as a form of in-situ data for group analysis, reserving our overall retrospective reflections for post-hoc interpretation during analysis and reporting.

3.5 Analysis

We analyzed data using a qualitative, inductive, thematic analysis across all of the data sources above [59]. First, we organized all materials chronologically and by type (student reflections, surveys, student work, instructor reflections). Each instructor independently conducted open coding to identify patterns, as well as using deductive analysis [45] to specifically identify semantic tensions and

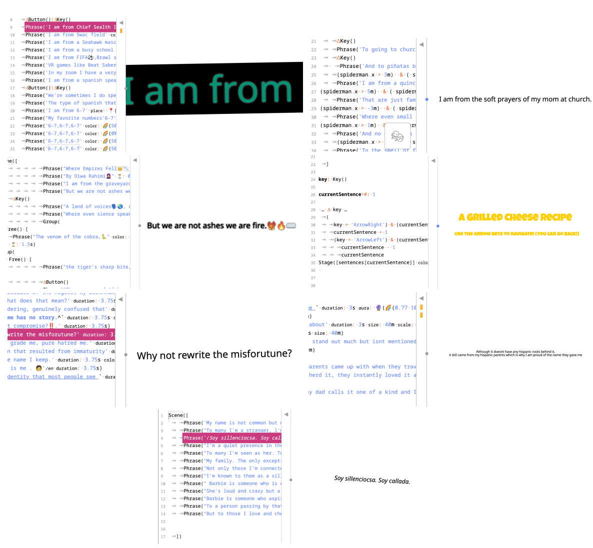


Figure 5: Wordplay projects from the class across the 6 weeks.

opportunities in the integration of LA and CS. After coding independently, we met as a team to compare interpretations and surface any points of disagreement.

Following the guidance of Hammer and Berland [26], we did not treat inductions and deductions as data, but rather as *claims* about data. Therefore, we treated disagreements not as a matter of reliability, but a point of inquiry to better form our claims. Much of our discussion about disagreements centered on what counted as genuine integration between LA and CS, as opposed to moments when students were experiencing the two disciplines in parallel. We also debated whether students' productive struggle with syntax, error messages, and unfamiliar coding constructs should be considered a barrier. The LA teacher viewed these challenges as a tension, while the CS teacher understood them as typical features of early programming practice. Through these conversations, we clarified our criteria for identifying tensions and ultimately decided not to treat these programming challenges as a thematic finding. We shared the key themes that emerged with the 4th author who was not involved in instruction, who took on a skeptical role, challenging whether claims were supported by data. The resulting themes presented in the results reflect this collaborative, qualitative sense-making process.

4 Results

Our study focused on two research questions: (1) what learning happens when we combine language arts and computer science learning in a transdisciplinary environment, and (2) what structural factors influenced those experiences. In the sections below, we share 6 themes that emerged in how students expressed their identities, communicated literary ideas through computational design, learned iteratively across creative writing and programming, and engaged in peer learning. We also describe constraints within the transdisciplinary structure, including limits on instructional time and the collaborative teaching needed to support work across disciplines. Throughout, we center on the ways that students synthesized across CS and LA in their expression and learning.

4.1 Multimodal Identity Expression

Throughout the course, students used Wordplay's multimodal features to show parts of their identity that their static text alone did not always capture. The multimodal features students used in this theme were implemented as inputs or parameters to Wordplay's built-in `Phrase` function. Students wrote code to specify values for attributes such as color, movement, size, opacity, emojis, and custom characters within the existing function structure. Students also incorporated their home languages into their Wordplay projects using the `Translate` feature, which allows programs and output to be machine translated into any combination of natural languages.

One student (*s1*) described how it felt to translate their poem into Spanish, as in Figure 6:

> "I was able to translate the text into Spanish very easily. I was born with two Mexican parents, so obviously that meant that my first language I ever learnt and spoke fluently was Spanish. I am a fluent speaker and I can write in Spanish ... [and] I think Wordplay

Figure 6: Wordplay output from (*s1*) showing a line of the poem translated into Spanish using the machine translation feature.

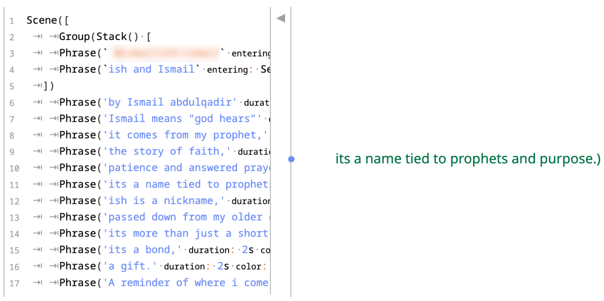


Figure 7: Wordplay output from s2 using green text to reference Islamic identity.

> giving me the option to translate my entire poem into Spanish made it easier to express myself in my poem."

Students incorporated color, animation, and imagery to reference personal and cultural identities. Several experimented with emojis and custom symbols using the `Characters` feature in Wordplay, which allowed students to create custom emojis and symbols to use in their projects and share with others. One student (*s2*) explained how moving their written poem into Wordplay prompted new forms of expression, as shown in Figure 7:

> "My name poem changed a lot when we switched to kinetic [when I moved my poem from paper into Wordplay] ... I added a Batman emoji to my poem, because I am a big Batman fan ... I also changed the color of the poem, I changed it to green because green is the color of Islam, and as a Muslim I put this in my name poem because my name comes from an Islamic prophet ... I was able to express myself better on Wordplay than the original."

Other students used Wordplay's multimodal features to express memories tied to family and culture. One student (*s2*) wrote in their graphic organizer that adding piñata emojis to their Wordplay project could remind them of "*my family's parties ... these sensory details showed and made me reflect on how I grew up with two different cultures.*"

Several students noted feeling more comfortable sharing about their identities when they could tailor how their stories appeared in a multimodal format through Wordplay. As one student (*s3*) explained, "*I could choose colors, animations, or layouts that I felt represented who I am. Both parts [the writing and the coding] worked together to give me a full way to express myself.*" Another student (*s4*) appreciated the options Wordplay gave them to "*change up the words and make them stand out or add feeling to it ... This helps readers get a better idea of the vibe my poem is about.*" A third student (*s3*) reflected that although sharing personal experiences felt vulnerable, presenting their work in Wordplay "*made me feel proud and accomplished ... I felt like my story mattered, and that made me feel seen.*"

Figure 8: Wordplay output stylized like Mexican films.

4.2 Literary Ideas Expressed through Code

Students made intentional design choices in their Wordplay projects to extend literary ideas from their creative writing. While students used some of the same multimodal features described in the previous theme, here they applied those features to communicate literary meaning within the structure of their projects.

Students implemented movement to express mood or tone, repetition or scale to draw attention to particular lines, or timing to communicate the pace of events. One student (*s5*) described using color and movement to elicit a sense of confusion from the reader, explaining that "*I added many different colors ... and movements helped emphasize the 'confusion' aspect.*"

Other students used computational features to deepen the sensory or emotional experience of their poems. One student (*s2*) planned user interactions that invited the audience to engage with the five senses:

> "I used the five senses in this poem. I wanted the audience to understand on a deeper level just how [I am] as a person ... For the Wordplay project I will use an emoji to show smell and put something to ask the audience 'oh what does this smell like,' and if they get it correct they move on."

Students also described how code supported literary visualization. One student (*s5*) wrote, "*Definitely with Wordplay I was able to add effects that helped get my point across further ... it also hit the audience of visual learners.*" Another student (*s3*) described choosing specific fonts, colors, and movements to reflect the look of classic Mexican movies she watched with her dad as a child, as in Figure 8. The project required users to click to advance through the poem, giving the reader a scene-by-scene reading experience.

4.3 Iterative Learning Across Language Arts and Computing

Students frequently described moving back and forth between writing and programming, with each process shaping the other. Written drafts often served as a starting point for later programming decisions, while planning code prompted revisions to writing. One student (*s4*) explained this relationship, referring to how the graphic organizers drawn from language arts supported their program planning:

> "I feel like the writing part is like a warm up or an idea of what I want to write and make it better for my own personal code ... the outlining techniques to structure my code helped define what my story was all about."

Students also drew parallels between revising writing and debugging code. One student (*s2*) noted "*the brainstorming between the writing and coding process*", describing moments when "*I had to think for 20 minutes whether I was writing or I was coding.*"

Instructor B described this iterative relationship as a "circle," noting that ideas formed through language arts gave students ideas for their programming, which were then "*enhanced by language arts, and brought to a new level of complexity students wouldn't reach for without LA.*" She explained that new opportunities in computing then influenced students' writing, "*leading to new creativity in their writing...and the circle continues.*"

Instructors also observed these cycles during peer review. During the final gallery walk for a project, Instructor A noted that students were simultaneously refining their writing and revising their code:

> "Students were soliciting both feedback on their writing as well as on the interactive effects added with Wordplay. Similarly, student feedback was offered both on elements of the writing as well as on Wordplay features that might be added in. Rather than just giving feedback on writing, or just giving feedback on coding, this final session was really an opportunity for both processes to be done together." (Instructor A, reflection log).

4.4 Peer Learning

Sharing identity-focused writing and Wordplay projects helped students build community and learn from one another. Several students noted that sharing helped them feel more connected. One student (*s4*) writing:

> "Sharing helped me build connections with my teachers and classmates ... which created a healthy, trustful community for everyone."

Another student (*s2*) explained that they found learning about peers' names, histories, and cultural backgrounds "*very awesome, because I want to get to know everyone in this class.*" A third student (*s5*) reflected that they felt more comfortable sharing their identity in this class than in previous settings:

> "I definitely felt more comfortable sharing more about my identity in this class since I usually never had the confidence to share more about my identity, especially to other new people I just met."

These relationships supported peer learning throughout the course. Students frequently sought ideas, feedback, and technical help from one another. One student (*s2*) wrote that they were "*the most excited person on earth to hear my peers' ideas of their Wordplay project,*" while another (*s6*) noted talking with friends to "*prepare for the poem and plan it out.*" Instructors also observed students helping each other debug code, incorporate new programming features, and offer specific feedback on both writing and Wordplay programming. During peer review, Instructor A noted in their reflection log that students "*were getting a good comfort level around sharing and giving/receiving feedback from each other,*" and that students were eager to receive and respond to feedback from peers.

Students' interest in each other's work also contributed to informal expertise sharing. Instructors detailed moments when students demonstrated how they programmed features such as background color, rainbow text, or grouping with classmates who wanted to incorporate similar ideas in their programming. These interactions often emerged spontaneously as students explored and discovered how to express their ideas in Wordplay. Instructor A reflected in their teaching log:

> "We know we don't have time to explicitly teach the programming for every possible input and output available in Wordplay, so we are building each day's plan in response to the things students want to learn to express what they want to express with their project... We encouraged the students to utilize each others' expertise and ask their peers to show them how they did something, if they see something they want to incorporate into their project too."

Building on these exchanges, instructors also noted that students relied on in person conversations and in the moment demonstrations. When a student discovered a feature or technique that others were interested in, instructors sometimes invited them, or students volunteered, to share their process with the class by presenting on the projector. The peer generated knowledge students shared in these moments was hard to capture or revisit later, yet it played an important role in how they learned from one another in real time. Instructor C reflected on the peer learning process, wondering "*to what extent this is an issue of deemphasizing our roles as the ones who hold knowledge vs how much classroom culture plays a role in students' comfort levels to seek help from one another*".

4.5 Shifts in Interest and Agency

Students related how their interest and confidence in programming grew as they developed their writing into Wordplay projects. Several students who had little or no prior experience with coding reported feeling more motivated to explore programming after the course. In their final feedback survey, one student (*s3*) wrote, "*I am proud of how it turned out, especially since I had never coded before,*" while another anonymous student explained "*I am really interested in programming... by doing it here I am really interested and I am looking forward to learning more about programming.*"

When asked about their main takeaway from the course in their final reflections, most students mentioned programming, describing new perspectives, increased interest, or a desire to continue learning.

Students connected these shifts to the transdisciplinary structure of the course. Several articulated how combining language arts and programming made coding feel more accessible and personally meaningful. One student (*s2*) reflected:

> "I really liked this class, and how it brought both language arts and programming into one. I never thought that the two could be a thing, but I guess it is, and I'm a fan of it. This brought me closer to programming and computer science (which I love)."

Another (*s4*) wrote that their "*relationship with both language arts and programming has changed throughout this course as I understand how coding programs work better and my love for writing poems has increased.*"

Several students reported feeling more confident as they worked through challenges in Wordplay. In the final feedback survey, one student anonymously explained that helping develop solutions to bugs "*helped me tremendously with thinking about how when things go wrong sometimes it's not your fault.*"

4.6 Barriers Within the Transdisciplinary Structure

While the previous themes highlight opportunities created through the integration of language arts and computing, students and instructors also noted barriers that influenced how the transdisciplinary structure played out in practice. The themes below focus on two key constraints: limited instructional time and the collaborative teaching capacity needed to support transdisciplinary work.

4.6.1 Time Constraints and Negotiating Learning Goals. Time constraints impacted how instructors balanced language arts and computing goals. Instructors described a need to continually negotiate and make deliberate choices about which learning goals to emphasize in each subject area, and which to set aside. Instructor B reflected:

> "rather than going in depth and teaching more computer science concepts like iterations and accessing items in a list, we all agreed that [moving forward with the project] was the best move given the remaining time that we have"

Instructor A similarly noted that:

> "with the limited time we have to accomplish our other course goals (including ELA goals) we can't do all the things we'd like to do if we had more time"

In several reflections, instructors reported choosing to prioritize foundational understandings, providing students with empowering first experiences with coding, and time for personal expression over introducing more advanced computer science concepts.

Students' creative ambitions often pushed against the time available, with students seeking more opportunities to revise their writing or Wordplay projects, noting that they wanted additional time to debug and add features. One instructor observed that a student had decided to abandon a more complex idea he was interested in implementing because the student worried they couldn't accomplish it within the available time. Instructor B reflected:

> "Students' writing process gave them new ideas for their Wordplay projects, and their coding process gave them new ideas for their writing. Students expressed a desire for more revision, editing, and debugging time for each project. They were very motivated to keep revising their work to meet their vision."

Instructors also noted scenarios in which the transdisciplinary structure created efficiencies rather than trade offs. During peer review, for example, students offered and received feedback on both writing and programming in the same session. Instructor A identified this as "*a good example of how integrating two subject areas doesn't always have to mean that there is less time for each subject,*" noting that students "*got the benefit of peer feedback on their writing and coding in the same amount of time it would have taken to do just one subject specific peer feedback session*".

4.6.2 Instructional Capacity and Transdisciplinary Collaboration. Supporting students across writing and programming relied on a collaborative teaching model that could be challenging to sustain. Instructors noted that the class felt most supportive when both content area teachers were present to support students and they moved between creative writing and coding. Instructor A wrote, "*It was really nice to have [both instructors] available to walk around and assist students...we really wanted to make sure that students experienced a feeling of success*". Instructors explained in reflection logs that supporting students in two disciplines added complexity, but that the small student-to-teacher ratio made it possible to offer individualized support when needed.

Students related the value of instructor support. Several noted in reflections that one-on-one check-ins with the instructors helped them refine their ideas, troubleshoot challenges, and move forward with both their writing and their programming. Instructors provided similar observations. Instructor B reflected, "*I am really glad for the times we can spend one-on-one with the students; I think it really helps give time for each specific idea each of them has*". Students also frequently named these check-ins as the most helpful part of each week in their weekly survey responses.

Instructors also described challenges of sustaining this collaborative environment that drew upon expertise in multiple disciplines, particularly when both subject area teachers could not be present. When the computer science teacher was absent, other instructors sometimes struggled to provide the support students were seeking around troubleshooting their code or understanding new computing concepts. In their reflections, Instructor C noted that this raised questions about "*what interdisciplinary teaching looks like when both content area teachers can't be present for every lesson*".

5 Discussion

Overall, our results provide several potential answers to our research questions. For RQ1, our results show that students learned introductory programming concepts about programs, program evaluation, functions, input, and output by using them to declaratively express interactive narrative. But they also learned about themselves, each other, and about the broader relevance of CS and LA in their lives and the world (Sections 4.1, 4.5). For RQ2, our results show that when creative writing and coding are taught together in a transdisciplinary, culturally responsive and sustaining Nicolescuian frame [42], adolescents can fluidly and jointly use both natural language and code as a medium for identity expression, about their ethnicities and lives (Section 4.1); they can engage CS as a new medium for literary ideas (Section 4.2), and they can learn about CS from each other through peer learning about writing, and about writing through peer learning about CS (Section 4.4). We also found that when taught as one, students did not view them as in disciplinary conflict, but rather as mutually supporting each other towards the goal of expression. While teachers observed this same mutuality, they also observed ways that disciplinary compromises were sometimes necessary to support students' learning and expression goals, but that in planning and feedback, disciplinary alignment between LA and CS actually created instructional efficiencies (Section 4.6). All instructors noted the importance of joint pedagogical content knowledge on the teaching team, and the importance of teacher-student ratio in more fully realizing these opportunities.

Our results broadly reinforce prior work on transdisciplinary learning and teaching [3, 23, 34, 42, 43]. Breaking down these particular disciplinary barriers did allow for a more holistic kind of learning and expression not possible in either LA or CS alone, with students moving fluidly between writing and programming, with each medium informing choices in the other, as well as catalyzing deeper learning in each discipline. Prior work on CS and LA integrations in primary learning have often found alignment in conceptual, mechanical, and motivational aspects of storytelling, [5, 9, 31, 32, 46, 54]; this study demonstrates similar structural alignments, while extending that work to adolescents, showing that the identity work possible in LA is also possible in CS learning when integrated in transdisciplinary ways. Whereas asymmetries in STEM and fine arts integrations, as well as CS and LA integrations in post-secondary, have been significant challenges [20, 36], we found that with one CS teacher and one LA teacher co-teaching in a medium focused on text, writing, and language, there was an abundance of opportunities to achieve balance. Moreover, that balance was frequently aligned with students' interests and learning goals. Additionally, like the many prior works on feedback in creative writing that have found the importance of rich peer feedback [62, 63], we also found that peer feedback was an essential element of students' motivation and learning. But we also found that the medium of code itself, at least in Wordplay's focus on text output, was itself a prompt for students' critical reflection on their writing and expression. Finally, our work shows one possible path for alleviating tensions between creative learning in secondary settings and emphasis on practical career-focused learning [17], by connecting practical programming skills to personally meaningful identity expression through creative writing.

While these results support the broader visions of transdisciplinarity [34, 42], our findings at the intersection of CS and LA are of course a proof of concept. It will be quite rare in any school setting globally for a CS and LA teacher to be able to co-teach, with an assistant, and focus intensely on the learning of only 7 adolescents. Moreover, the summer school setting of this study further separates it from the realities of the school year, where demands on student attention and the stakes of learning are heightened. Our choice of Wordplay, as opposed to other programmable media also with similar storytelling capacity, also leaves open questions about how much of our results were a product of the medium, versus pedagogy or other factors. Finally, this study did not reckon with the often rigid constraints placed on secondary CS and LA learning by learning standards, uniform curricula, and other expectations by school leaders, families, and communities. Our work was not intended to be generalizable in a direct way to these many complex settings, but rather to examine the possibilities of schools re-imagined around transdisciplinarity, revealing what benefits such new visions might offer, and what further questions we might need to answer to assess the viability of those benefits.

Given these limitations, our findings raise many open research questions. Are similar synergies between CS and creative writing for adolescents possible in other programmable media, such as *Processing*, *Scratch*, or even professional general purpose languages like *Python*, *Java*, and *Typescript*, or do media and tools matter crucially, as Peppler has observed in the context of STEAM learning [47]? Can solo teachers be prepared to teach in transdisciplinary ways across CS and creative writing, or is co-teaching an essential model for transdisciplinary teaching? What constraints do the school year, larger class sizes, and curricular constraints pose, and can those constraints be meaningfully overcome, or even leveraged, to promote similar transdisciplinary learning? And what are the long term learning benefits of identity expression by adolescents in CS and LA settings, beyond the interest, inclusion, and motivation benefits we observed in the classroom? These and other research questions are essential to imagining a future of schools that is not only supportive of youth diversity, but also reflective of the inherently transdisciplinary world that youth enter as they leave school [43].

Ultimately, our results show that if we can answer these questions, the kinds of learning that schools and teachers might enable for adolescents are far more feasible than we might have imagined. Teachers might take these results and wonder, "*Maybe I should talk to my colleagues, and see if they're interested in trying something similar.*" Courageous school leaders might experiment with bringing their LA and CS teachers together to explore the possibilities of bringing computation in conversation with creative writing. Secondary teacher educators in both LA and CS might make visible, or even emphasize, that disciplinary approaches to these subject areas are not the only viable ways to teach, or even the best ways to teach. These explorations might help continue to demonstrate what most computer scientists already know: that computing can be and often already is an essential part of most disciplines, and it's about time we figured out how to reflect that in schools and student learning.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 2417014, 2031265, 2100296, 2122950 and unrestricted gifts from Microsoft, Adobe, and Google. Thank you to the University of Washington Upward Bound staff, to the Center for Learning, Computing, and Imagination for their feedback on this draft, and to Siddhu Bhimireddy for his rich perspectives on creative writing and CS.

References

[1] Omar H. Abed. 2016. Drama-Based Science Teaching and Its Effect on Students' Understanding of Scientific Concepts and Their Attitudes Towards Science Learning. *International Education Studies* 9, 10 (2016), 163–173. doi:10.5539/ies.v9n10p163

[2] Terry Anderson and Julie Shattuck. 2012. Design-based research: A decade of progress in education research? *Educational Researcher* 41, 1 (2012), 16–25. doi:10.3102/0013189X11428813

[3] Chris Argyris and Donald A. Schön. 1974. *Theory in Practice: Increasing Professional Effectiveness.* Jossey-Bass.

[4] Susan Bailey and R. Watson. 1998. Establishing Basic Ecological Understanding in Younger Pupils: A Pilot Evaluation of a Strategy Based on Drama/Role Play. *International Journal of Science Education* 20, 2 (1998), 139–152. doi:10.1080/0950069980200202

[5] Marina Umaschi Bers. 2019. Coding as another language: A pedagogical approach for teaching computer science in early childhood. *Journal of Computers in Education* 6, 4 (2019), 499–528. doi:10.1007/s40692-019-00147-3

[6] Carla Brodley, Valerie Barr, Ellen Gunter, Mark Guzdial, Ran Libeskind-Hadas, and Bill Manaris. 2024. CS+ X—Challenges and Opportunities in Developing Interdisciplinary-Computing Curricula. *ACM Inroads* 15, 3 (2024), 42–50. doi:10.1145/3687131

[7] Carla E. Brodley, Benjamin J. Hescott, Jessica Biron, Amanda Ressing, Michael Peikin, Sarah Maravetz, and Alan Mislove. 2022. Broadening Participation in Computing via Ubiquitous Combined Majors (CS+X). In *ACM Technical Symposium on Computer Science Education (SIGCSE)*. ACM. doi:10.1145/3478431.3499352

[8] Ann L Brown. 1992. Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings. *The journal of the learning sciences* 2, 2 (1992), 141–178.

[9] Quinn Burke and Yasmin B. Kafai. 2010. Programming & storytelling: Opportunities for learning about coding & composition. In *ACM International Conference on Interaction Design and Children (IDC)*. Association for Computing Machinery, 348–351. doi:10.1145/1810543.1810611

[10] C. Cabo and Reneta D. Lansiquot. 2016. Integrating Creative Writing and Computational Thinking to Develop Interdisciplinary Connections. In *CUNY Academic Works*. New York City College of Technology. doi:10.18260/p.25795

[11] J. E. Calabrese, Nancy B. Songer, Jr. Aina, D. K., et al. 2025. Re-examining the Relationship Between Three-Dimensional Science and Engineering Knowledge and Student Interest: A Mixed Methods Study. *Disciplinary and Interdisciplinary Science Education Research* 7, 17 (2025). doi:10.1186/s43031-025-00134-y

[12] Meg Callahan and Jennifer M. King. 2011. Classroom Remix: Patterns of Pedagogy in a Techno-Literacies Poetry Unit. *Journal of Adolescent & Adult Literacy* (2011). doi:10.1002/JAAL.00016

[13] Linda Christensen. 2000. *Reading, Writing, and Rising Up: Teaching About Social Justice and the Power of the Written Word*. Milwaukee: Rethinking Schools.

[14] Quentin M. Clark. 2025. A Pedagogical Approach: Toward Leveraging Mathematical Modeling and AI to Support Integrating Humanities into STEM Education. *Frontiers in Education* 9 (2025). doi:10.3389/feduc.2024.1396104

[15] G. Clifton. 2021. Critical-creative literacy and creative writing pedagogy. *University of Toronto Quarterly* 91, 1 (2021), 51–66. doi:10.3138/utq.91.1.04

[16] Susan M. Drake and Joanne L. Reid. 2018. Integrated Curriculum as an Effective Way to Teach 21st-Century Capabilities. *Asia Pacific Journal of Educational Research* 1, 1 (2018), 31–50. doi:10.30777/APJER.2018.1.1.03

[17] Chris Drew. 2016. Minding the Pedagogical Gap: Creative Writing Studies, Common Core Standards, and the Secondary Creative Writing Moment. *Journal of Creative Writing Studies* 2, 1 (2016). https://repository.rit.edu/jcws/vol2/iss1/3

[18] Ron Eglash, Juan E Gilbert, Valerie Taylor, and Susan R Geier. 2013. Culturally responsive computing in urban, after-school contexts: Two approaches. *Urban Education* 48, 5 (2013), 629–656.

[19] Jayne Everson, F Megumi Kivuva, Camilo Montes De Haro, and Amy J Ko. 2026. Culture-centric computational embroidery as a medium for learning computer science. *Computer Science Education* (2026), 1–34.

[20] Lindsay Finch, Claudia Moreno, and R. Benjamin Shapiro. 2021. Luminous Science: Teachers Designing for and Developing Transdisciplinary Thinking and Learning. *Cognition and Instruction* 39, 4 (2021), 512–560. doi:10.1080/07370008.2021.1945064

[21] Geneva Gay. 2002. Preparing for Culturally Responsive Teaching. *Journal of Teacher Education* 53 (2002), 106–116. doi:10.1177/0022487102053002003

[22] Hannah R. Gerber and Debra P. Price. 2011. Twenty-First-Century Adolescents, Writing, and New Media: Meeting the Challenge with Game Controllers and Laptops. *The English Journal* 101, 2 (2011), 68–73. http://www.jstor.org/stable/41415429

[23] Charles Goodwin. 1994. Professional Vision. *American Anthropologist* 96, 3 (1994), 606–633. doi:10.1525/aa.1994.96.3.02a00100

[24] Kelly W. Guyotte, Nikki W. Sochacka, Tricia E. Costantino, Joachim Walther, and Nadia N. Kellam. 2014. STEAM as Social Practice: Cultivating Creativity in Transdisciplinary Spaces. *Art Education* 67, 6 (2014), 12–19. doi:10.1080/00043125.2014.11519293

[25] Mark Guzdial and Tamara Nelson-Fromm. 2025. Designing Courses for Liberal Arts and Sciences Students Contextualized around Creative Expression and Social Justice. In *ACM Technical Symposium on Computer Science Education (SIGCSE)*. 423–429. doi:10.1145/3641554.3701896

[26] David Hammer and Leema K Berland. 2014. Confusing claims for data: A critique of common practices for presenting qualitative research on learning. *Journal of the Learning Sciences* 23, 1 (2014), 37–46. doi:10.1080/10508406.2013.802652

[27] Danah Henriksen. 2013. Full STEAM Ahead: Creativity in Excellent STEM Teaching Practices. *The Transdisciplinary STEAM Journal* 1, 2 (2013). doi:10.5642/steam.20140102.15

[28] Leslie R. Herrenkohl and Joseph L. Polman. 2018. Learning Within and Beyond the Disciplines. In *International Handbook of the Learning Sciences*, Frank Fischer, Cindy E. Hmelo-Silver, Susan R. Goldman, and Peter Reimann (Eds.). Routledge/Taylor & Francis, New York, NY, 106–115.

[29] Glynda A. Hull and Mira-Lisa Katz. 2006. Crafting an agentive self: Case studies of digital storytelling. *Research in the Teaching of English* 41, 1 (2006), 43–81. doi:10.58680/rte20065995

[30] Zuzana Husárová. 2012. A Method of Teaching Multimodal Creative Writing. *Studies in Foreign Language Education* 4 (2012), 1–20.

[31] Sharin Rawhiya Jacob, Miranda C. Parker, and Mark Warschauer. 2022. *Integration of computational thinking into English language arts.* Association for Computing Machinery, New York, NY, USA, 55–63. doi:10.1145/3507951.3519288

[32] Caitlin Kelleher and Randy Pausch. 2007. Using storytelling to motivate programming. *Commun. ACM* 50, 7 (2007), 58–64. doi:10.1145/1272516.1272540

[33] William Kist. 2005. *New Literacies in Action: Teaching and Learning in Multiple Media.* Teachers College Press, New York.

[34] G. Kligyte, J. Melvold, S. Pratt, and B. Bowdler. 2025. Educating for sustainable and equitable futures: A transdisciplinary future-making capability framework. *Futures* 155 (2025), Article 103665. doi:10.1016/j.futures.2025.103665

[35] Amy J. Ko, Anne Beitlers, Brett Wortzman, Matt Davidson, Alannah Oleson, Mara Kirdani-Ryan, Stefania Druga, and Jayne Everson. 2023. *Critically conscious computing: methods for secondary education.* Self-published. https://criticallyconsciouscomputing.org/

[36] Reneta D. Lansiquot (Ed.). 2016. *Interdisciplinary Pedagogy for STEM: A Collaborative Case Study* (1 ed.). Palgrave Macmillan, New York. XVIII + 143 pages. doi:10.1057/978-1-137-56745-1 eBook ISBN 978-1-137-56745-1..

[37] Anna Leahy. 2005. *Power and Identity in the Creative Writing Classroom: The Authority Project*. Multilingual Matters & Channel View Publications. http://www.jstor.org/stable/jj.29308427

[38] Timothy L. Lynch. 2017. How English Teachers Will Save the Future: Reimagining Computer Science as the Language Art It Is. *STEM Journal* 18, 4 (2017), 163–180. doi:10.16875/stem.2017.18.4.163

[39] Tia C Madkins and Nicol R Howard. 2023. Engaging culturally relevant and responsive pedagogies in computer science classrooms. *Computer Science Education: Perspectives on Teaching and Learning in School* 101 (2023), 101–113.

[40] Mary B. McVee, Nancy M. Bailey, and Lynn E. Shanahan. 2008. Teachers and Teacher Educators Learning from New Literacies and New Technologies. *Teaching Education* 19, 3 (2008), 197–210. doi:10.1080/10476210802250216

[41] R. J. A. Metcalfe, S. Abbott, P. Bray, J. Exley, and D. Wisnia. 1984. Teaching Science through Drama: An Empirical Investigation. *Research in Science & Technological Education* 2, 1 (1984), 77–81. doi:10.1080/0263514840020109

[42] Basarab Nicolescu. 2002. *Manifesto of Transdisciplinarity*. State University of New York Press. Original work published 1996.

[43] Richard K Norton, Elisabeth R Gerber, Paul Fontaine, Gail Hohner, and Patricia D Koman. 2022. The promise and challenge of integrating multidisciplinary and civically engaged learning. *Journal of planning education and research* 42, 1 (2022), 102–117. doi:10.1177/0739456X18803435

[44] Christos H. Papadimitriou. 2003. Mythematics: storytelling in the teaching of computer science and mathematics. In *Conference on Innovation and Technology in Computer Science Education*. ACM, 1. doi:10.1145/961511.961513

[45] Noel Pearse. 2019. An illustration of deductive analysis in qualitative research. In *European Conference on Research Methodology for Business and Management Studies*. 264. doi:10.34190/JBRM.17.3.004

[46] Emrah Pektas and Florence R. Sullivan. 2021. Storytelling through programming in Scratch: Interdisciplinary integration in the elementary English language arts classroom. In *Asia Pacific Society for Computers in Education International Conference on Computational Thinking and STEM Education*. 1–5.

[47] Kylie Peppler and Naomi Thompson. 2024. Tools and materials as non-neutral actors in STEAM education. *Journal of the Learning Sciences* 33, 4-5 (2024), 719–756. doi:10.1080/10508406.2024.2380694

[48] Ane Portillo-Blanco, Jenaro Guisasola, and Kristina Zuza. 2025. Integrated STEM education: addressing theoretical ambiguities and practical applications. *Frontiers in Education* 10 (2025). doi:10.3389/feduc.2025.1568885

[49] Cassie F. Quigley, Danielle Herro, and Faiza M. Jamil. 2017. Developing a Conceptual Model of STEAM Teaching Practices. *School Science and Mathematics* 117, 1–2 (2017), 1–12. doi:10.1111/ssm.12201

[50] Jennifer Rowsell and Maureen Walsh. 2011. Rethinking literacy education in new times: Multimodality, multiliteracies, and new literacies. *Brock Education*

Journal 21, 1 (2011), 53–62. doi:10.26522/brocked.v21i1.236

[51] Jean J Ryoo. 2019. Pedagogy that supports computer science for all. *ACM Transactions on Computing Education (TOCE)* 19, 4 (2019), 1–23.

[52] Emmanuel Schanzer, Kathi Fisler, and Shriram Krishnamurthi. 2018. Assessing Bootstrap: Algebra students on scaffolded and unscaffolded word problems. In *ACM Technical Symposium on Computer Science Education (SIGCSE)*. 8–13. doi:10.1145/3159450.3159498

[53] Kimberly A Scott, Kimberly M Sheridan, and Kevin Clark. 2015. Culturally responsive computing: A theory revisited. *Learning, media and technology* 40, 4 (2015), 412–436.

[54] Shaunna Smith and Lauren E. Burrow. 2016. Programming multimedia stories in Scratch to integrate computational thinking and writing with elementary students. *Journal of Mathematics Education* 9, 2 (2016), 119–131.

[55] B. C. Stagg. 2019. Meeting Linnaeus: Improving Comprehension of Biological Classification and Attitudes to Plants Using Drama in Primary Science Education. *Research in Science & Technological Education* (2019), 1–19. doi:10.1080/02635143.2019.1605347 (online first).

[56] B. C. Stagg and M. F. Verde. 2019. Story of a Seed: Educational Theatre Improves Students' Comprehension of Plant Reproduction and Attitudes to Plants in Primary Science Education. *Research in Science & Technological Education* 37, 1 (2019), 15–35. doi:10.1080/02635143.2018.1455655

[57] Annelies Struyf, Hans De Loof, Jelle Boeve-de Pauw, and Peter Van Petegem. 2019. Students' Engagement in Different STEM Learning Environments: Integrated STEM Education as Promising Practice? *International Journal of Science Education* 41, 2 (2019), 1–21. doi:10.1080/09500693.2019.1607983

[58] Dustin C. Summey. 2013. *Developing Digital Literacies: A Framework for Professional Learning*. Corwin.

[59] Gareth Terry, Nikki Hayfield, Victoria Clarke, Virginia Braun, et al. 2017. Thematic analysis. *The SAGE handbook of qualitative research in psychology* 2, 17-37 (2017), 25.

[60] Tiffany Tseng, Matt J Davidson, Luis Morales-Navarro, Jennifer King Chen, Victoria Delaney, Mark Leibowitz, Jazbo Beason, and R Benjamin Shapiro. 2024. Co-ML: Collaborative machine learning model building for developing dataset design practices. *ACM Transactions on Computing Education* 24, 2 (2024), 1–37.

[61] Monique Verhoeven, Astrid MG Poorthuis, and Monique Volman. 2019. The role of school in adolescents' identity development. A literature review. *Educational psychology review* 31, 1 (2019), 35–63. doi:10.1007/s10648-018-9457-3

[62] Ryan S. Wooley, Christopher A. Was, Christian D. Schunn, and David W. Dalton. 2008. The Effects of Feedback Elaboration on the Giver of Feedback. In *Annual Meeting of the Cognitive Science Society*. Washington, DC.

[63] Fuhui Zhang, Christian D Schunn, and Alok Baikadi. 2017. Charting the routes to revision: An interplay of writing goals, peer comments, and self-reflections from peer reviews. *Instructional Science* 45, 5 (2017), 679–707. doi:10.1007/s11251-017-9420-6