Choreographing Increased Understanding and Positive Attitudes towards Coding By Integrating Dance

Breeann FLESCH[1]

Camila GABALDÓN[1]

Matthew NABITY[1]

Darryl THOMAS[1]

[1]Western Oregon University

DOI: 10.21585/ijcses.v4i3.109

Abstract

Increasing the inclusion of underrepresented individuals in coding is an intractable problem, with a variety of initiatives trying to improve the situation. Many of these initiatives involve STEAM education, which combines the arts with traditional STEM disciplines. Evidence is emerging that this approach is making headway on this complex problem. We present one such initiative, iLumiDance Coding, which attempts to pique the interest and increase confidence of students in coding, by combining it with a fun and physical activity: dance. The connections between dance and coding, while not immediately obvious, are authentic, and we hypothesize that this approach will increase student comfort level with coding. We used student surveys of attitudes toward coding and a variety of statistical approaches to analyze our initiative. Each analysis showed a positive effect on student comfort level with coding. These results are useful for both educators and researchers since they contribute to a deeper understanding of how to increase interest in coding, which we hope will lead to an increase in representation.

Keywords: STEAM education, coding, increasing representation, dance, coding

1. Introduction

Finding ways to spark interest in groups not immediately drawn to coding has been a challenge in an economy that consistently lacks candidates and diversity for positions that require coding skills. Women and minorities are consistently underrepresented in these areas and several initiatives have emerged to attract both of those groups, with mixed results. It is a problem that begins before students even enter college, often starting in middle school. Youth may determine career paths by the age of 13 (Tai et al., 2006; Bernstein et al., 2019; Shet & Tremblay, 2019). Furthermore, the literature addressing middle school students' engagement and identification with STEM and STEM - related careers indicates that developing STEM-specific individual interest (Staus et al., 2020) and preserving feelings of self - efficacy in STEM during middle school is crucial (Aschbacher et al., 2014; Barmby, Kind, & Jones, 2008; Kang et al., 2019; Haussler & Hoffmann, 2002; Nugent et al., 2015). Partovi (2016) of Code.org reports that high school girls and underrepresented minorities who take AP Computer Science are more likely to major in computer science than those who do not (girls 11% vs 1%; underrepresented minorities 14-17% vs 2%). Increasing participation, though, is not simply a matter of offering coursework. In this paper, we present *iLumiDance and Coding*, which attempts to pique the interest and increase confidence of students in coding, by combining it with a fun and physical activity: dance.

Creating engagement by using music and coding have been explored in a variety of projects and has been shown to be a powerful way to teach and learn computer science content, as well as increase coding self-efficacy (Barate et al., 2017; Bell & Bell, 2018; Horn et al., 2020; Ruthmann et al., 2010). Combining dance and coding has been explored to a lesser extent. Shamir et al. (2019) saw positive results from the MathDance, which allowed student to perform a dance while learning computing concepts but did not explicitly draw on the similarities between dancing and coding. While not a combination that immediately comes to mind, explicitly teaching dance and coding together is a powerful way to create authentic cross-disciplinary connections and spark the interest of learners who might otherwise be reluctant in either domain. Upon closer examination, it becomes clear that the ties between dance are coding are numerous and authentic. Coding is a language, just as the construction of visual and kinetic images are a "language" in dance choreography. The word choreography literally means "dance-writing" from the Greek words for dance and writing. In the most basic sense, coding is a set of instructions given to a computer for execution. Dance choreography parallels coding in that it provides instructions for the human body, a "computer" that is self-aware, conscious. Beyond just providing instructions, dance choreography and coding share many concepts that cross both disciplines. Both choreography and coding are, at their core, sequences of primitive instructions that are executed in a linear order. They both utilize repetition, whether in an “ABA” ternary form or a “for loop.” They also have conditional statements that help them respond to dynamic situations or states. For example, the choreography may instruct one dancer to execute a solo movement sequence until within a foot of a second dancer, after the music reaches a certain point both dancers lock arms and revolve around a central point or swing your partner round and round. The moonwalk made popular by the late Michael Jackson, provides an excellent mirror to the use of “if-then” statements in coding. The choreography may instruct the dancer to moonwalk until reaching the edge of the performance space, then execute a 540-degree clockwise spin and continue to moonwalk in the opposite direction. Finally, in both coding and choreography we create “functions” by naming sets of instructions or dance moves (e.g. running man, waltz, polka, twerking, electric slide), which can then be called time and again, without having to recreate the instructions from scratch.

With all of these authentic connections, we conjecture that teaching students coding through dance will increase their comfort level with coding.

2. Literature Review

In recent years, the economy has shown an ever-increasing need in STEM fields, for which the number of qualified applicants is consistently insufficient (Yankman, 2008). In addition to an insufficient number of applicants, employers are also seeking creativity and innovation in new hires that may require them to pull knowledge from more than the discipline in which an employee might be trained (Land, 2013). STEAM education, which expands traditional STEM education to include the arts as a way to foster creative thinking, has emerged as an important pedagogy and may help meet these needs. As employers have noted, complex problems often require using a multi- or trans-disciplinary approach, which is implicit in STEAM education. The process of integrating arts and sciences encourages learners to demonstrate adaptive critical thinking as they hone their ability to develop flexible problem-solving models. The representation of subjects from both the arts and sciences as equals in STEAM education inspires open-ended creative exploration and serves as a form of productive play and inquiry (Trowsdale, 2016).

The interest in STEAM has further increased with the release of the Next Generation Science Standards (NGSS) in 2013, as these standards are designed to teach students to “think critically, analyze information, and solve complex problems” (NGSS, 2013). One NGSS that can be addressed specifically through STEAM projects with coding components is “Using Mathematics and Computational Thinking.” Computational thinking is an approach to problem solving that is based on the fundamentals of computing, and this approach is transforming the way we think about many disciplines, including the natural sciences (Bundy, 2007).

The idea of crosscutting concepts, concepts that are applicable across the domains of science, are also a core part of the NGSS (NGSS, 2019). In many STEAM applications, crosscutting between the arts and sciences is added. Science the language and concepts around science topics can be intimidating to students (Graham and Brouillette, 2016). Likewise, many students are initially intimidated by coding concepts, so, presenting them in a way that shows their connection to the arts has been shown to increase students' comfort level with coding (Moore et al., 2016).

STEAM education not only creates cross-disciplinary connections that can lead to greater creativity, but it also can increase retention of content in within-discipline content. Graham and Brouillette (2016) found that the introduction of arts content into STEM education in elementary schools increased test scores in science areas by 13 percentiles compared to the control group. It has also been shown that by adding the "A" to STEM, we can bring more underrepresented communities into STEM, which continues to be an intractable problem. According to the National Center of Science and Engineering Statistics (NCSES, 2019), of the bachelor's degrees awarded in Computer Science in 2016, only 18.7% were awarded to women and 21.6% were awarded to underrepresented minorities. There are no doubt many causes of this complex problem, but one survey found that women do not think the STEM fields are as interesting as other fields (Weinberger, 2005). A powerful approach to combat this can be through integrating STEM into fields in which girls and women are already interested. For example, STEAM projects that emphasize aesthetics and creativity have been shown to increase participation by young women and girls (Buechley and Mako Hill, 2010). Buchholdz, Shivley, Peppler, and Wohlwend (2014) discovered in their mixed-gender dyads that the "traditional" gender scripts for computing were disrupted by replacing traditional circuitry materials with conductive thread, fabrics, and needles, providing girls the opportunity to take on leadership roles in completing highly complex electronics projects. Another study showed an increase in interest in STEM in a Native American population by using culturally responsive STEAM projects (Kant, Burckhard and Meyers, 2018). Plus, integrating arts into STEM can make the STEM fields more approachable to all, including underrepresented groups (Maltas, 2015).

The use of STEAM projects to learn coding and electronics is not an entirely new concept. For example, the EarSketch project, which integrates music with coding, found that "results reveal that students' attitudes positively and statistically significantly increased across all constructs in our Student Engagement survey, which included constructs such as computing confidence, computing enjoyment, computing perceived usefulness, motivation to succeed in computing, identity and belonging in computing, and intention to persist" (Moore et al., 2016). Several other programs have also shown strong positive results when connecting music to computer science (Barate et al., 2017; Bell & Bell, 2018; Horn et al., 2020). The Performatics program for college students described by Ruthmann et al. (2010), utilized scratch to teach music students coding concepts and computational thinking while they remixed and created musical pieces. Shamir et al. (2019) showed an increased interest in STEM classes and careers after students combined, music, dance, and animation with computer science. Peppler (2013) took another STEAM approach and utilized e-textiles with students. As part of this, they noted that "Stitching circuits seem to demystify ideas that can be elusive to students using traditional electronics."

3. Methods

While there are multiple STEAM approaches that can be used to introduce coding, our project focuses specifically on the use of dance. It involves the delivery of curriculum utilizing "puppets" and costumes that light up, laptops with our software installed and a connection between them, which currently runs via WiFi over a local router. Students receive some dance and coding instruction but are allowed plenty of room for exploration and creativity in what they design. Our project is founded in the constructivist theories of education, which dates back to Dewey (1934) and even earlier thinkers. In a constructivist framework, students are encouraged to explore, create and play to construct their own understanding of the world. STEAM education is well aligned to fulfill the tenets of constructivist education, where "students are engaged and central to knowledge productions" (Gross and Gross, 2016).

In 2013, Peppler introduced "eight guiding principles of STEAM-powered computing education," most of which we utilize in our curriculum and delivery. These principles are:

1. Choose open-ended, personal, and aesthetic tools and materials,
2. Make design thinking central,
3. Create authentic combinations of STEM and the arts,
4. Facilitate easy-entry, but challenging, designs,
5. Purposefully contrast multiple media, tools, and materials,
6. Involve a range of disciplinary experts,
7. Devise new assessments, pedagogy, and learning environments,

8. Document and showcase work.

As we described in the introduction, the connections between dance and code are many and authentic. During the course of our STEAM instruction, we explicitly reiterate these authentic connections as we are working on projects, meeting the third principle. Our software is based on Google's Blockly. The colorful, block-based interface provided by Blockly alleviates some of the "intimidation factor" of coding and allows for easy entry into basic coding concepts, while leaving plenty of room for challenge and creativity, meeting the fourth principle. Students spend most of their time working with the coding software, the puppets and the choreography, with some short, direct instruction. For example, students explore the use of functions and loops in their code on their own, using these tools to develop their desired performance look. This method also follows Peppler’s second principle, "make design thinking central." Students are doing this when they try some code, test it to check the visual impact, and then use that as feedback to further adjust their code. They are designing a visual impact in collaboration with designing movement, all the while constructing knowledge about the elements of coding and computational thinking.

Computational thinking, an important 21st century skill, is a core part of the curriculum. And, according to Wing (2018), "the essence of computational thinking is abstraction." In our curriculum, students get experience with abstraction in a variety of ways. For example, students will work on a function, say called lightingChase, where the lights turn on and off in rapid succession. Once they get lightingChase just right, then they can call that function as many times as needed, creating a layer of abstraction. Furthermore, students need to grapple with the abstraction of presentation - between the computer screen and the physical puppets or costumes.

3.1 Research Design

Our curriculum is flexible; we can teach a 5-hour workshop, a week-long camp or even a 10-week course with our software, and it can be adapted for students as young as 4th grade all the way up to young adults. In the 5-hour workshop, students will learn the basics of coding, including the sequential nature of executing code and simple iteration through looping. They will then perform a dance that is choreographed ahead of time but will design the lighting that accompanies the dance using their coding skills. In longer courses, students also learn about conditional statements and functions. They dive deeper into the fundamentals of movement and expression through choreography and develop their own dance(s) with accompanying lights.

When possible, our entire content delivery team is present, which consists of a librarian and professors of dance and computer science, each of whom brings disciplinary expertise and perspective that is unique. This follows Peppler’s (2013) sixth principle, "Involve a range of disciplinary experts" and, while one of us will typically be the leader of the workshop, all of us assist and may present pieces as our content areas are discussed.

In the workshop format, the students are split into six groups of six to program six puppets. Within those groups, each student is assigned a laptop to code on and a body part (head, torso, right arm, right leg, left arm, and left leg), which they will program to light up using Blockly. Each part has six differently colored EL (electro-luminescent) wires that can be programmed as solid, flickering, or off and two LED strips that can produce a custom color based on RGB values the student picks. The laptops and puppets are connected to a local network via a home router system. The EL wires and LED strips are controlled via WiFi using a 12 channel WiFi DC dimmer. For much of the workshop, the students focus on learning how to control the different lights using individual commands, loops, and functions that they design in Blockly. Blockly then produces JavaScript, which both runs the puppets on the laptop screens and sends commands to the individual puppet pieces and wires via the routers. At the end of the workshop, they are asked to coordinate both the movement and lighting of their puppet piece to music in conjunction with their table-mates to create a performance. This follows Peppler’s (2013) eighth principle "Document and showcase work."

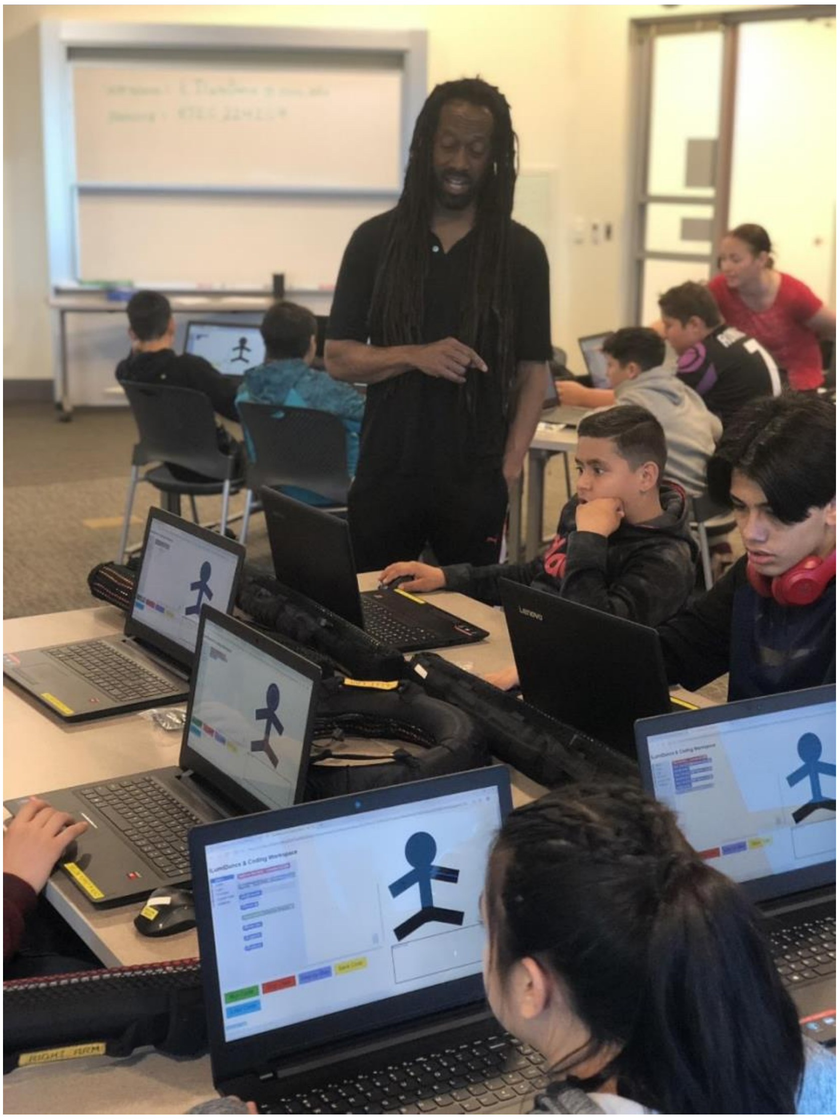

Figure 1. Two team members working with students

In the summer camp, students designed a costume, with separate EL wires in each appendage, which are also controlled via the code they design in Blockly. Like the puppets, the costumes are controlled using a WiFi DC dimmer, but instead of controlling six individually hued wires, each channel controls a single wire representing each body part. The students are asked to bring dark colored clothing, which they use as their costume base. Each is given six pieces of EL wire and six pieces of mesh that they can thread the EL wire through in a design pattern that they choose before attaching the mesh to the appropriate part of their costume. Students then design a lighting sequence and choreograph dancing using Blockly. They may then combine their choreography and lighting for a group dance with shared lighting effects that may appear to create motion or other effects.

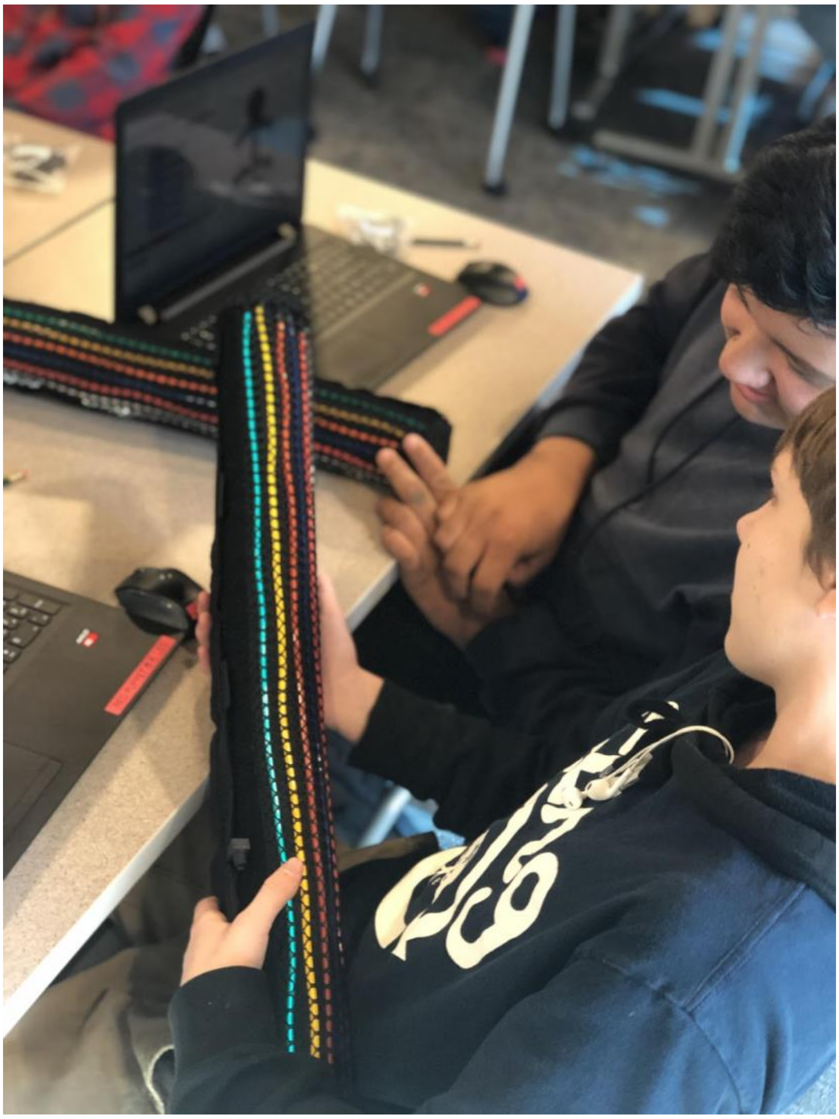

Figure 2. Students programming a puppet piece during a workshop

The 10-week course is still in development but will incorporate additional detailed lessons on coding and movement and the connections between them. Because it is a college-level course, students will also be researching a social issue of their choosing and developing a dance with coded lighting to highlight their representation of that issue. Readings will include articles and chapters on topics such as cultural intersections of dance, art, communication, and technology.

3.2 Data Collection

After each course, we distribute a survey on the workshop to the participants that have given, or whose guardians have given, informed consent, based on our institutional review board approval. Because of limited time, we gave the survey once, at the end of the workshop, and then we chose to maintain a consistent survey method for the summer camp. Other data collection methods, like exit interviews or focus groups, where not used in order to maximize the amount of time for the activity itself, as these students had a strict schedule be get bussed back to their home school. This survey asks workshop participants questions about their attitude about coding before, during and after the survey. For example, they are asked "After participating in the workshop, my comfort level with coding is…." and asked to circle numbers on a Likert scale 1 through 5, with 1 being associated with 'Lower', 3 being associated with 'The Same' and 5 being associated with 'Higher.' Since the 5-hour workshops are for students, there are often other educators in the room (their classroom teachers, paraprofessional educators, counselors, etc.). When this is the case, we also give them a survey that measures their perspective of computing before, during and after the workshop. For example, one question is "I plan to incorporate more coding into my education environments”, and they are asked to circle a number on a Likert scale of 1 through 5 with 1 associated 'Not at all,' 3 associated with 'Maybe,' and 5 associated with 'Most Definitely.' These surveys were based on Heidi’s (2013) work with measuring effectiveness of Injecting Computational Thinking into Computing Activities for Middle School Girls, which allowed us to use an existing, validated tool.

3.3. Participants

The workshop from which we collected data had students from Prineville, Oregon in Crook County. According to the Oregon Department of Education, 61% of the students from this district are on free and reduced-price lunch, 16% of the students identify as Hispanic and 6% as multiracial (https://www.ode.state.or.us/data/reportcard/reports.aspx). The summer camp pulls students from the Salem-Keizer School District, which has 71% of students on free and reduced-price lunch, 40% identify as Hispanic and 5% as multiracial (https://www.ode.state.or.us/data/reportcard/reports.aspx).

4. Results

The results indicated that combining coding and dance is an effective way of increasing interest, efficacy, and engagement in coding. Participants reported increased confidence with coding, feelings of creativity, playfulness and engagement after the workshops or summer camps.

4.1 Data Analysis

The survey results were used to test our hypothesis that combining dance with coding has positive effect on student comfort level with coding. The questions that we used to measure this comfort level were:

- Question 3: "After participating in the workshop, my comfort level with coding is…"
- Question 4: "If I choose to, in the future I can do a job that uses coding."
- Question 5a: "When I worked on coding the puppets, I felt confused"
- Question 5b: "When I worked on coding the puppets, I felt creative"
- Question 5c: "When I worked on coding the puppets, I felt playful"
- Question 5d: "When I worked on coding the puppets, I felt bored"

Summaries of the data from the spring workshop and the summer camp are reported in Table 1 and Table 2 respectively, and all data is aggregated in Table 3. In addition to reporting the actual data for the survey questions (3, 4, 5a, 5b, 5c, 5d), we compared the survey results in each to randomly completed surveys. As a first assumption for comparison, we considered each of the five responses to be equally likely so that the underlying distribution is uniform. This means the probability of observing any individual outcome from {1,2, 3, 4, 5} is ⅕. Using this, we defined a success as follows:

X= success if we observe a 4 or a 5 on questions 3, 4, 5b, 5c,

or X= success if we observe a 1 or a 2 on questions 5a, 5d,

which establishes underlying binomial distributions for the various values of n, the number of survey respondents. From a probabilistic point of view, this means we should expect 40% of the survey results in the respective categories we define as a success. The results in Table 1 suggest that success rate is much higher than predicted for all but Question 4. Looking at the estimated standard deviation in each case, $\sigma_X \approx 2.3$ and $\sigma_X \approx 2.4$ respectively, the observed number of successes is noticeably higher for all but Question 4 as the observed results are nearly two standard deviations or more above what theory predicts. This suggests either the underlying distribution is not uniform or that there was a positive effect on student comfort level with coding from the workshop. Similar trends are apparent in the summer camp data and the aggregated data.

Likert data in the context of education, specifically STEM disciplines, has received renewed interest (Brickman and Lovelace, 2013). A typical assumption for Likert data is that the data is normally distributed. *"Likert (1932)* *himself argued that the distances of scores such as 1, 2, 3, 4, 5 are equal and yield data which are approximately normally distributed. The question is if the same is true for the labels (e.g. 'Strongly Agree') which are frequently used. Recent research suggests that literacy affects the ability to discriminate between categories, i.e. that the suitability of the classical Likert scale depends on the choice of the sample (Chachamovich et al., 2009)."* (Treiblmaier and Filzmoser, 2009).

We revisited the calculations in Table 1-3 using various approximate normal distributions. Comparing the expected number of successes with the observed data in Tables 1-3, we found that the observed number of successes was now higher than expected for all questions. The theoretical standard deviations decrease as well making the observed data stand out a bit more than before. Assuming approximate normal distributions indicates the positive effect from the workshop or summer camp was more noticeable. To investigate the possible effect from the workshop and summer camp, we now turn to an inferential analysis of the data to test the hypothesis that the interventions had positive effects.

Table 1. Spring Workshop Summary Survey Results

Question	Predicted # of Successes	Observed # of Successes	n	$\frac{x}{n}$(%)	$\|X_{obs} – X_{pred}\|$	Difference from Predicted (%)
3	9.2	17	23	73.9	7.8	33.9
4	9.6	8	24	33.3	1.6	6.7
5a	9.6	14	24	58.3	4.4	18.3
5b	9.6	16	24	66.7	6.4	26.7
5c	9.2	17	23	73.9	7.8	33.9
5d	9.6	17	24	70.8	7.4	30.8

Table 2. Summer Camp Summary Survey Results

Question	Predicted # of Successes	Observed # of Successes	n	$\frac{x}{n}$(%)	$\|X_{obs} – X_{pred}\|$	Difference from Predicted (%)
3	5.6	8	14	57.1	2.4	17.1
4	5.6	5	14	35.7	0.6	4.3
5a	5.6	6	14	42.9	0.4	2.9
5b	5.6	12	14	85.7	6.4	45.7
5c	5.6	11	14	78.6	5.4	38.6
5d	5.6	9	14	64.2	3.4	24.3

Table 3. Spring Workshop Summary Survey Results

Question	Predicted # of Successes	Observed # of Successes	n	$\frac{x}{n}$(%)	$\lvert X_{obs} - X_{pred} \rvert$	Difference from Predicted (%)
3	14.8	25	37	67.6	10.2	27.6
4	15.2	13	38	34.2	2.2	5.8
5a	15.2	20	38	52.6	4.8	12.6
5b	15.2	28	38	73.7	12.8	33.7
5c	14.8	28	37	75.7	13.2	35.7
5d	15.2	26	38	68.4	10.8	28.4

4.1.1 One-sided Hypothesis Test

We begin with a simple null hypothesis $H_0: p = 0.4$ and test this against the claim that the workshops had a positive effect so that the alternate hypothesis is $H_1: p > 0.4$ for each of the six questions. Based on our initial analysis, we opted to assume the underlying distribution is uniform and to use the test statistic $Z = \frac{\frac{x}{n} - p_o}{\sqrt{\frac{p_0(1-p_0)}{n}}} \geq z_\alpha$, where α is the approximate level of the test. The results for the six questions along with the computed p-values are given in Table 4. For Questions 3, 5b, 5c, and 5d, we note a statistically significant result for nearly any typical level of significance. Additionally, the p-value for question 5a is significant depending on the desired level of significance.

Table 4. One-sided alternate hypothesis

Question	p_0	x	n	$\hat{p}$	$Z_{\text{test statistic}}$	p-value
3	0.40	25	37	$\frac{25}{37} = 0.676$	3.42	0.00031
4	0.40	13	38	$\frac{13}{38} = 0.342$	-0.730	0.77
5a	0.40	20	38	$\frac{20}{38} = 0.526$	1.59	0.056
5b	0.40	28	38	$\frac{28}{38} = 0.737$	4.24	0.000011
5c	0.40	28	37	$\frac{28}{37} = 0.757$	4.43	0.0000047
5d	0.40	26	38	$\frac{26}{38} = 0.684$	3.58	0.00017

4.1.2 Two-sided Hypothesis Test

The preceding calculations and analysis used hypothesis tests found in almost any statistics text. In practice, these standard confidence intervals and tests of hypotheses often have poor performance even when the sample size is rather large (Agresti and Caffo, 2000). Agresti and Coull (1998) suggested the "add two successes and two failures" interval given by $\tilde{p} \pm z_{\frac{\alpha}{2}}\sqrt{\frac{\tilde{p}(1-\tilde{p})}{\tilde{n}}}$, where $\tilde{n} = n + 4$ and $\tilde{p} = \frac{X+2}{n+4}$. Careful analysis has shown that this interval performs dramatically better than the standard Wald intervals and their associated hypothesis tests explored earlier in this analysis, even when working with small samples (Agresti and Coull, 1998). This confidence interval corresponds to the test statistics $|Z| = \left|\frac{\tilde{p}-p_0}{\sqrt{\frac{p_0(1-p_0)}{\tilde{n}}}}\right| \geq z_{\frac{\alpha}{2}}, or\ |Z| = \left|\frac{\tilde{p}-p_0}{\sqrt{\frac{\tilde{p}(1-\tilde{p})}{\tilde{n}}}}\right| \geq z_{\frac{\alpha}{2}}$, where the difference in the denominator is somewhat up to the practitioner. Using the latter of these two we compute the results reported in Table 5.

Table 5. Two-sided alternate hypothesis

Question	p_0	x	n	$\tilde{p}$	$Z_{\text{test statistic}}$	p-value
3	0.40	25	37	$\frac{25+2}{37+4} = 0.659$	3.49	0.00048
4	0.40	13	38	$\frac{13+2}{38+4} = 0.357$	-0.58	0.56
5a	0.40	20	38	$\frac{20+2}{38+4} = 0.524$	1.61	0.11
5b	0.40	28	38	$\frac{28+2}{38+4} = 0.714$	4.51	0.0000065
5c	0.40	28	37	$\frac{28+2}{37+4} = 0.732$	4.79	0.0000017
5d	0.40	26	38	$\frac{26+2}{38+4} = 0.667$	3.67	0.00024

In any of the preceding calculations, the results are consistent across the inferential structures. Questions 3, 5b, 5c, and 5d are all statistically significant indicating there is evidence for increased confidence with coding, feelings of creativity, playfulness and engagement after the treatment. The Question of student confusion depends on the desired level of significance but at the traditional level of $\alpha = 0.05$ the results are significant. The effect of combining dance and coding is statistically relevant for certain key features central to student comfort level and merit further study.

5. Conclusions

STEAM education is growing rapidly as an important approach to teaching and retaining students in STEM, especially coding (Graham and Brouillette, 2016). Teaching coding with a STEAM approach increases the excitement and creativity around coding and brings underrepresented groups into STEM (Barate et al., 2017;

Bell & Bell, 2018; Horn et al., 2020; Ruthmann et al., 2010, Shamir et al. 2019). However, we feel it is important that the STEAM connections to coding be authentic and transdisciplinary, which is why coding and dance are such a natural fit.

The numerous connections and parallel concepts between coding and dance make it a natural and authentic pairing for instruction. It is approachable to people with a variety of backgrounds or comfort levels with either activity and promotes creativity and playful exploration of both. In each of the venues where we collected data, we measured students' attitudes towards coding as a possible measure of future success as described by Else-Quest, Mineo, and Higgins (2013) and in each case, we found that there was increased comfort with coding, feelings of creativity, playfulness and engagement using this approach. Like Wanzer et al. (2020), we saw an increased feeling of affiliation toward coding and coding activities in students self-reporting after participation in the activities. In addition to the statistical data we gathered, we received lots of enthusiastic verbal feedback from participants. Of particular note, one 8th grade girl commented that "I really was expecting us to do some normal coding, like we do at school. But, this was totally different. It’s more hands-on. And I think if we did more of this, more girls would take coding" after the workshop.

While this outcome can't help but be exciting, we must acknowledge some of the limitations in our study. First, our sample sizes are small, and we did not have a control group. Our project is in its early stages, so the data must be viewed as preliminary. Second, our samples, while diverse, were not entirely randomly selected; the summer camp participants self-selected and the workshop participants had teachers who had selected to involve their classes with these activities, which could indicate previous exposure to coding and coding concepts, though does not account for changes in attitude.

Our future work includes using this approach in a quarter-long class at the college level and attempt to measure any change in the understanding of coding basics in addition to attitudes. We will also continue to provide workshops and summer camps and collect data on student attitudes. We hope to start collecting data on race and gender, and as our sample gets bigger, disaggregate the data to further explore underrepresented groups.

Over the course of this project we have met with a variety of individuals from classroom teachers to large tech company executives, and many expressed frustrations about the difficulty of sparking excitement for coding. Based on our results, our project provides one avenue to address that frustration by combining learning coding with a topic that some already feel passionate about. This approach to teaching coding can be implemented by K-8 teachers as well as informal science educators. While our workshops utilized puppets and dedicated laptops, the approach does not require these. Instructors can utilize existing software and platforms, such as Scratch or Blockly and still emphasize the authentic connections between coding and dance or another activity that combines similar concepts. An example of this might be creating artwork, finger weavings, or music based on coding concepts. Explicitly describing and utilizing computer science concepts, such as iteration, looping, and variables, while doing that activity, then showing how that concept is also used in the context of creating code can make powerful connections in students’ minds. The efficacy of pairing coding and dance compared to other combinations of artistic endeavors and coding is unknown, but there may be an advantage to having a variety of STEAM approaches to getting individuals excited about coding. We are enthusiastic about the inroads into coding that these STEAM projects provide and are interested in how the results of other STEAM pairings compare to ours.

References

Agresti, A. and Coull, B.A. (1998). Approximate is better than ‘exact’ for interval estimation of binomial proportions. *The American Statistician*, 52, p. 119126.

Agresti, A. and Caffo, B. (2000). Simple and effective confidence intervals for proportions and differences of proportions result from adding two successes and two failures, *The American Statistician*. 54, 280288.

Aschbacher, P. R., Ing, M., & Tsai, S. M. (2014). Is science me? Exploring middle school students’ STEM career aspirations. *Journal of Science Education and Technology*, 23(6), 735–743. https://doi.org/10.1007/s10956-014-9504-x

Baratè, Adriano, Luca A. Ludovico, and Dario Malchiodi. "Fostering computational thinking in primary school through a LEGO®-based music notation." *Procedia computer science*, 112 (2017): 1334-1344.

Barmby, P., Kind, P. M., & Jones, K. (2008). Examining changing attitudes in secondary school science. *International Journal of Science Education*, 30, 1075–1093.

Bell, J., & Bell, T. (2018). Integrating computational thinking with a music education context. *Informatics in Education*, 17(2), 151-166.

Brickman, P. and Lovelace, M. (2013). Best practices for measuring students' attitudes toward learning science. *CBE life sciences education*. 4. 606617.

Buchholz, B., Shively, K., Peppler, K., and Wohlwend, K. (2014). Hands On, Hands Off: Gendered Access in Crafting and Electronics Practices. *Mind, Culture, and Activity.* (21)4, 278-297. https://doi.org/10.1080/10749039.2014.939762

Buechley, L. and Mako Hill, B. (2010). LilyPad in the wild: how hardware's long tail is supporting new engineering and design communities. *In Proceedings of the 8th ACM Conference on Designing Interactive Systems (DIS '10).* ACM, New York, NY, USA, 199-207. 10.1145/1858171.1858206

Bundy, A. (2007). Computational thinking is pervasive. *Journal of Scientific and Practical Computing*. 1(2), 67-69

Dewey, J. (1934). *Art as experience.* Westport, CT: Praeger.

Else-Quest, N., Mineo, C., and Higgins, A. (2013). Math and Science Attitudes and Achievement at the Intersection of Gender and Ethnicity. *Psychology of Women Quarterly.* (3)37, 293-309. https://doi.org/10.1177/0361684313480694

Gross, K. and Gross, D. (2016). TRANSFORMATION: Constructivism, Design Thinking, and Elementary STEAM, *Art Education,* (6)69, 36-43. https://doi.org/10.1080/00043125.2016.1224869

Haussler, P., & Hoffmann, L. (2002). An intervention study to enhance girls' interest, self-concept, and achievement in physics classes. *Journal of Research in Science Teaching*, 39(9), 870–888.

Horn, M., Banerjee, A., West, M., Pinkard, N., Pratt, A., Freeman, J., ... & McKlin, T. (2020). TunePad: Engaging learners at the intersection of music and code.

Kang, H, et al. (2019). How do middle school girls of color develop STEM identities? Middle school girls' participation in science activities and identification with STEM careers. *Science Education,* 103:418–439.

Kant, J., Burckhard, S. and Meyers, R. (2018). Engaging High School Girls in Native American Culturally Responsive STEAM Activities. *Journal of STEM Education*, 18(5). Retrieved September 13, 2019 from https://www.learntechlib.org/p/182466/

Maltas, W. (2015). From STEM to STEAM: Integrating Arts Education into the STEM disciplines of Science, Technology, Engineering and Math. Drexel University, Philadelphia, PA.

Moore, R., Edwards, D., Freeman, J., Magerko, B., McKlin, T., and Xambo, A. (2016). EarSketch: An Authentic, STEAM-based Approach to Computing Education. 2016 American Society for Engineering Education Annual Conference & Exposition.

NGSS Lead States, (2013). *Next generation science standards: For states, by states.*

Nugent, G., Barker, B., Welch, G., Grandgenett, N., Wu, C., & Nelson, C. (2015). A model of factors contributing to STEM learning and career orientation. *International Journal of Science Education*, 37(7), 1067–1088.

Oregon Department of Education. 2019. At-A-Glance School and District Profiles. https://www.ode.state.or.us/data/reportcard/reports.aspx Accessed 28 September 2019.

National Center of Science and Engineering Statistics. https://ncses.nsf.gov/pubs/nsf19304/digest/field-of-degree-women#computer-science. Accessed 28 September 2019

Peppler, K. (2013) STEAM-Powered Computing Education: Using E-Textiles to Integrate the Arts and STEM, *Computer*, 46(9), 38 - 43. DOI: 10.1109/MC.2013.257.

Ruthmann, A., Heines, J. M., Greher, G. R., Laidler, P., & Saulters, C. (2010, March). Teaching computational thinking through musical live coding in scratch. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 351-355).

Shamir, M., Kocherovsky, M., & Chung, C. (2019, March). A paradigm for teaching math and computer science concepts in k-12 learning environment by integrating coding, animation, dance, music and art. In *2019 IEEE Integrated STEM Education Conference (ISEC)* (pp. 62-68). IEEE.

Treiblmaier, H. and Filzmoser, P. (2009). Benefits from using continuous rating scales in online survey research, tech. rep., Vienna University of Technology, Augasse 2-6, A-1090 Vienna, Austria, November2009. https://pdfs.semanticscholar.org/6f90/c913e1e5a162371862a6c5e20f8e7de5e9c7.pdf

Wanzer, D. L., McKlin, T., Freeman, J., Magerko, B., & Lee, T. (2020). Promoting intentions to persist in computing: an examination of six years of the EarSketch program. *Computer Science Education*, 1-26.

Webb, H. (2013). *Injecting Computational Thinking into Computing Activities for Middle School Girls, A Dissertation in Information Sciences and Technology from The Pennsylvania State University*. The Graduate School College of Information Sciences and Technology.

Weinberger, C.J. (2004). Just ask! Why surveyed women did not pursue IT courses or careers, *IEEE Technology and Society Magazine* 23, 28-35.

Wing J.M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.

Yakman, G., (2008). STEAM Education: An Overview of Creating a Model of Integrative Education, Virgina Polytechnic and State University: Virgina. https://www.iteea.org/File.aspx?id=86752&v=75ab076a