Teaching a Computer to Sing:
A Middle School, After-School Pilot Program Integrating Computing and Music

Jesse M. Heines
Dept. of Computer Science
Univ. of Massachusetts Lowell
1-978-251-9350
Jesse_Heines@uml.edu

Daniel A. Walzer
Dept. of Music
Univ. of Massachusetts Lowell
1-978-934-3881
Daniel_Walzer@uml.edu

Rachel R. M. Crawford
Bartlett Comm. Partnership School
Lowell (MA) Public Schools
1-978-937-8968
RCrawford@lowell.k12.ma.us

Jill H. Lohmeier
Graduate School of Education
Univ. of Massachusetts Lowell
1-978-934-4617
Jill_Lohmeier@uml.edu

Shanna R. Thompson
Graduate School of Education
Univ. of Massachusetts Lowell
1-978-934-4641
Shanna_Thompson@uml.edu

ABSTRACT
Music is universal. Music is also mathematical, and mathematics is of course the basis of computing. This paper reports on the first year of our attempt to create a voluntary after-school program that introduces middle school students to computing through music. As expected, we had successes and failures. This paper details our goals, activities, changes made in mid-year, evaluation results, and plans for the future.

CCS Concepts
• **Social and professional topics → K-12 education.**

Keywords
Computing and Music; Middle School; After-School.

1. FROM SINGING TO PROGRAMMING
It probably goes without saying that most educators, regardless of field, think it's important for kids to be computer literate and even to learn at least a little about how to program. Of course we agree. But how do we "hook 'em," especially in an after-school program that doesn't even begin until after they've been in school for seven hours and are ready to just play and hang out with their friends?

"There's nothing like making music and messing with sound to inspire people to learn how to program." [25]

In a nutshell, our approach in this project has been to work with a dynamic music teacher whom the children adore, have her teach them songs with multiple parts and modest complexity, and then have the children manipulate recordings of those songs in Audacity [1] and program them in Scratch [20] and Pencil Code [16]. They then "perform" their creations for each other using a computer projector and the room's sound system, thereby enjoying an outlet for their work and learning from what others have produced.

ACM SIGCSE 2017, March 8-11, 2017, Seattle, WA, USA.

DOI: http://dx.doi.org/10.1145/12345.67890

While this approach may sound straightforward, it wasn't easy to implement. Despite their love of listening to—and singing along with—popular music with their friends, the kids in our program were initially shy about singing in a class setting. The music teacher worked hard to get them to overcome their reluctance. The team struggled to engage them in the computer part of the program, too. Due to our lack of experience with this age group, it took some time for us to "get our footing." However, as discussed in this paper, persistence paid off and the children made significant progress in both singing and programming.

2. ENABLING RESEARCH
We believe that our current research is unique in that it explores the intersection of computing and music at the middle school level. The basis for this research is our prior work in Performamatics [14, 15]. The first of our prior projects explored interdisciplinary connections between computer science and three fine arts fields: art, music, and theater [5, 7, 8, 9, 13, 18]. It then focused on computing+music [8, 19], which for us turned out to be the collaboration that had the most "traction." The popularity of the university courses we built in those projects and the positive research results we observed enticed us to see if we could achieve similar gains with younger students. Other research related to ours is discussed below.

Teaching a Computer to Sing builds on the premise outlined by Magerko et al. [12] that creative coding can enhance musicianship by helping students identify a song's structure. It also builds on the concept of "coding through play" described by Stinson [21], who taught Scratch coding by having children follow robots through a series of imaginative storytelling activities. The key here is that children saw immediate, tangible results as the robots responded to various commands. Programming music also yielded immediate results, which children learned to identify as "right" or "wrong" by what they heard.

Ho [10] conducted a five-year study to evaluate the ability of information technology to inspire learner-centered music creation practices in young students. Both students and teachers remarked that the use of MIDI improved pitch and rhythmic accuracy when added to traditional choral rehearsal techniques and pedagogy. Our work builds on this finding by having students work with the same music in multiple modes.

Studer [22] explored the use of computers in choral rehearsals and noted that music notation programs are highly useful for isolating each part. We took an analogous approach: having children write programs that "sang" multiple parts (see Section 3.2) to help them learn each one. Studer also found that music notation programs enhance STEM learning along with musical competency. Our project went one step further: writing computer programs to play music using standard computing constructs (see Section 3.1).

3. PROGRAM LOGISTICS

Teaching a Computer to Sing is an after-school program. It ran on Tuesdays and Thursdays from October 13, 2015, through May 26, 2016, on the following schedule:

2:10-2:25 snacks
2:25-3:20 singing with the school music teacher
3:20-3:50 break for homework and snacks
3:50-4:45 computing with university faculty and student assistants
4:45-5:00 dinner.

Students spent the first afternoon session singing songs and then, after a break, spent the second coding those songs using ABC Notation [2, 26] in Pencil Code [16], a block programming environment. Sometimes we used the last 10-15 minutes of the computing session for children to show their work to the entire group. This yielded mixed results, as some were enthusiastic to show their creations while others were too self-conscious to do so or were simply ready to go home. Also, we often felt it was best not to disturb them if they were fully engaged in the day's activities.

3.1 Software Choices

We began the program using Scratch as our music platform, but the middle school students had a lot of trouble working with MIDI. First, converting musical scores to MIDI involved two steps: (a) figuring out a note's alphabetic name (such as C) from the staff, and then (b) converting that note to its MIDI value (such as 60). Second, the resulting code bore little resemblance to the musical notation. For example, it's pretty hard to know that the code in Figure 1(a) plays *Frère Jacques* even if one compares it to the sheet music in Figure 1(b). The Pencil Code version in Figure 1(c) that uses ABC notation still requires a bit of interpretation, but it is clearly easier to explain to students than the Scratch version.

To help students make the transition from sheet music to ABC notation we used WebMusicScore [6]. This is a wonderful free website that allows one to enter ABC notation in one window and see the resultant score in another window. Students can thus compare the resultant score with the original score that they have on paper to ensure that they are the same. If the two scores are not identical, the ABC notation is easily edited to correct any problems.

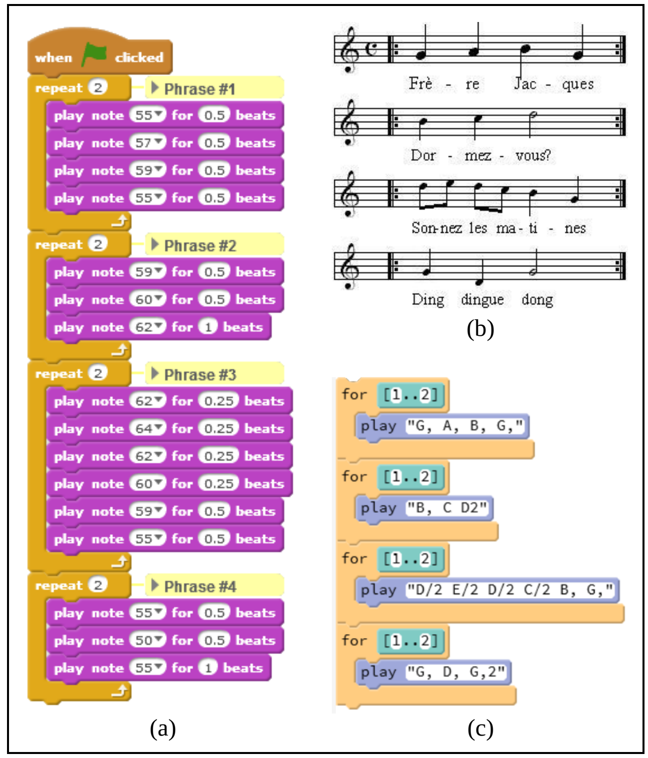

Figure 1. *Frère Jacques* in (a) Scratch, (b) standard music notation, and (c) Pencil Code using ABC Notation.

Once the correct ABC notation is entered into WebMusicScore, it is easy to copy and paste that into Pencil Code. The final step is then to enclose the ABC notation in Pencil Code in double quotes and add code to pass it to a function that plays it. "Pure" Pencil Code using the built-in `play` function is shown in Figure 1(c), but a little additional code displayed a dynamic keyboard that could show which piano key was being played as each note sounded. We wrote a custom function called `sing` that built on this enhanced capability to allow students to play up to four parts simultaneously simply by specifying which phrase to play on which piano.

The code in Figure 2 (shown in text rather than block mode) calls our custom `sing` function to play *Frère Jacques* as a round (in the key of C). Note that line 10 rests part 2 for 8 beats before "singing" that part, thus creating the round. The graphic in Figure 2 shows the keyboards as they are displayed while the music is playing.

3.2 Song Choices

Songs were chosen by the school music teacher in conjunction with the two university professors. We began the program using popular songs such Rachel Platten's *Fight Song* [17], Taylor Swift's *Shake It Off* [23], and Shawn Mendes's *Stitches* [4]. We had an arranger create simple harmony parts for these songs, but that did not prove as popular with the students as we expected. The children had trouble with the harmony parts because the rhythms of these songs are complex.

The music teacher on our team suggested that we switch to "partner songs," which were sets of three simple, complementary melodies meant to be sung together. One such set consisted of *One Bottle of Pop*, *Don't Throw Your Trash in My Backyard*, and *Fish and Chips and Vinegar*. [Go to youtu.be/u-TdsmPHjo0 to hear these songs sung individually and then together, but not by our students.] It was much easier for the students to sing and code these simpler songs in multiple parts.

```
for [1..2]    # part 1, phrase 1
  sing 1, “C D E C”
for [1..2]    # part 1, phrase 2
  sing 1, “E F G2”
for [1..2]    # part 1, phrase 3
  sing 1, “G3/4 A/4 G/2 F/2 E C”
for [1..2]    # part 1, phrase 4
  sing 1, “C G, C2”

sing 2,”Z8”   # rest for 8 beats
for [1..2]    # part 2, phrase 1
  sing 2,”C D E C”
for [1..2]    # part 2, phrase 2
  sing 2,”E F G2”
for [1..2]    # part 2, phrase 3
  sing 2, “G3/4 A/4 G/2 F/2 E C”
for [1..2]    # part 2, phrase 4
  sing 2, “C G, C2”
```

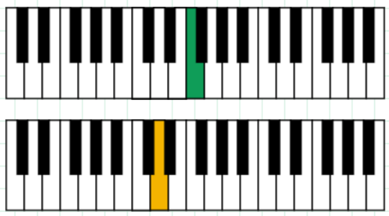

Figure 2. **Playing *Frère Jacques* as a round.**

3.3 Student Assistants

One of the program’s key features was the relatively large number of university student assistants we employed. We had originally budgeted for two assistants, but it quickly became apparent that that was not enough. We really needed one university student for each pair of middle school students, so we increased the number of assistants to six.

To establish rapport, the university students (and professors) sang with the middle schoolers as well as assisted them both with reading music (which was necessary to translate scores into MIDI or ABC notation) and actual coding. About half the assistants were music majors, while the other half were computer science majors.

Unfortunately, however, we had only one female assistant: Nicole. With 12 female and 2 male children in the program, we feel that it is important that we work harder to recruit more female assistants next year, especially since Nicole provided us with invaluable insights about the complex issues that middle school girls deal with and thus helped us weather a number of storms.

As with any program that uses student assistants, some worked out well and some did not. Most were good at helping to keep the children on task and getting them past hurdles such as computer freeze-ups and simple programming issues. Some made excellent suggestions during our activities planning sessions, and some provided insightful observations when we reviewed each day’s experiences.

Some assistants even established strong personal relationships with the middle schoolers and functioned as role models, which contributed significantly to the clubhouse atmosphere we were trying to maintain. When one assistant was absent, the children were always disappointed and asked if they would be there the next time we met.

3.4 Additional Performances

Another aspect of the program involved bringing in others involved in both music and computing to perform for the students. Our student assistants who were music majors occasionally performed, but we also brought in one of the *a cappella* groups on campus to demonstrate multipart singing.

Our arranger came by one day so that the children could meet the person who had created the arrangements they were singing. He taught them a new, simple multipart song, and they were simply enthralled by his ability to sing each of the different parts and not get confused!

4. RESEARCH QUESTIONS AND FIRST YEAR FINDINGS

Our research investigated two instructional questions and two supporting questions. We attempted to answer these questions both by documenting our own informal observations and by employing surveys and focus groups conducted by the external evaluation team. The former obviously suffer somewhat from observer bias, while the latter suffer somewhat from small sample size. (We started the year with 18 students and ended with 14.) Nonetheless, we feel that we can draw a number of conclusions from our first year experiences that can be reasonably supported by credible anecdotal evidence if not by hard numbers.

4.1 Instructional Questions

(1) *Can middle schoolers follow the connections from singing to digitized sound to MIDI or ABC notation and back to music to help them learn to program using the songs they like to sing?*

The answer to this question is an unqualified “yes,” and we would go even further to say “yes, and sometimes with enthusiasm.” Of course, not every student was “into” the coding part of the program, and some days it seemed like none of them wanted to do any coding at all. (As every parent knows, such is life when working with children.) But on other days a good number of students exhibited real excitement about their ability to “teach a computer to sing” and eagerly lined up for “show and tell” at the end of the day to demonstrate their accomplishments to others.

We began the year having students code music as a series of play blocks. As the year progressed, we introduced progressively more advanced computing concepts. The first of these was simple loops (like those in Figure 2), which allowed students to replay musical phrases that repeated in succession. Next we introduced loops with control variables, such as for k in [1..3]. This allowed students to make the connection between songs with larger repetitive structures that used first and second endings and conditional (if) statements in code. We then introduced general variables, which allowed us to store and reuse musical phrases coded as ABC notation strings. A couple of students even got as far as indexed variables (one-dimensional arrays), which we used as two parallel structures to pair each note with its lyric.

One student in our program was a 7th grader, while all others were 5th and 6th graders. Only one had prior experience with computer programming, and 4 of the 14 did not have a computer at home. Given these demographics, the fact that the students had already been in school for more than eight hours by the time they began programming with us, and our own inexperience in teaching middle school students, we feel that the programming concepts we were able to introduce represent a reasonable level of learning.

(2) *Conversely, can programming their individual parts help students learn to sing in three- and four-part harmony?*

During initial discussions, the music teacher told the professors that she thought it would be difficult to get the children singing in more than two parts because they had never done it before and had no familiarity with it. By the end of the year, however, they were successfully singing the "partner songs" (Section 3.2) in three parts. They even cheered when they all finished at the same time! We can't give all the credit for that progress to the students' work in programming those songs, but the music teacher felt that at least some credit was due there.

One of the programming techniques that seemed to help students learn multipart songs was the use of variables to store and reuse musical phrases coded as ABC notation strings. This helped students see song structures, notice where phrases repeated, and understand how the melody lines went together. Again we do not want to overstate this result as it was impossible to measure objectively, we but have learned to trust the music teacher's instincts, as she is clearly highly attuned to the students' capabilities.

4.2 Supporting Questions

(3) *What resources, models, and tools (RMTs) are necessary to integrate STEM education into a middle school after-school choral program?*

We found the public school resources to be severely limited. The computer network and access to the Internet are so severely tied down that Windows systems could not access the network at all and sites such as YouTube were not accessible without teacher credentials. Luckily, access to all the music sites mentioned earlier was available to us.

We were also unable to install software on the school systems, and no one in the school had authority to do so, either. We had to make a request of the central school district office, and that took weeks to be fulfilled. To resolve this issue, we are planning to buy systems specifically for the project's exclusive use.

Our model for the after-school program was initially what we were used to: a laboratory class. This approach did not prove viable, as the students were simply unable to pay attention to instructions for more than a minute or two in the after-school environment. We therefore transitioned toward a clubhouse model, where students worked one-on-one or two-on-one with a professor, university student assistant, or another middle school student.

We prepared handouts with instructions and illustrations so that they could work on their own rather than listen to us explain how to accomplish the day's goals. We also hired three times the number of university student assistants than we had planned to, as it became evident that they were needed in the clubhouse model.

The tools we used have been discussed previously, but it is important to reemphasize that they changed and rotated throughout the year. The switch from Scratch to Pencil Code was the biggest unanticipated change at the beginning of the year, and the discovery of WebMusicScore for writing ABC notation proved a godsend.

(4) *Can the involvement of older students and teachers who match the students' racial and/or cultural backgrounds have a positive effect on the "people like me don't (or can't) do that" belief that so often stifles efforts to attract underrepresented groups to STEM?*

One of the issues that concerned us was our ability to "connect" with the middle school students. Almost all of the children had very different racial profiles from our own, and, as noted in Section 3.3, the vast majority were female (86% of 14 total). Numerous authors such as Kohl [11], Delpit [3], and Tatum [24] have written about issues related to race in the classroom, and we feared that at least some of those issues extended to gender, as well.

Looking back, it appears that we need not have been so concerned. While almost all of the children were black or of Asian or Hispanic or mixed descent and the professors and assistants were mostly white males, this did not appear to be a major stumbling block. The children did build relationships faster with some of the university students than with the professors, so age might also have been a factor here. We simply do not know whether the relationships would have developed more rapidly or deeply with mentors who more closely matched the children's racial profiles.

Gender seemed to play a larger role in relationship-building, as we observed at least some of the female children gravitating more freely toward Nicole, our lone female assistant, than toward the males. Then again, Nicole is of Hispanic descent, so that may have influenced the children's feelings toward her, as well. That said, the children also gravitated toward working with one of the white male assistants who exhibited exceptional teaching and interpersonal skills. With time, however, we feel that each of us was eventually able to "connect" to each child.

The bottom line is that although we can only report observational and anecdotal evidence, we feel that, in our case, the premise of this question appears to be unsupported. That said, we acknowledge that our project has a small sample size and that our results are dependent on the many specific personalities involved. Thus, it is difficult—if not impossible—to generalize these results.

4.3 Lessons Learned So Far

Based on these experiences, here are the main lessons we feel that we have learned so far:

- Working with children after they've been in school for 8 or more hours is hard. There are times when one has to just let them play. In addition, one must understand that some days they just won't want to code, and that pushing them to do so is futile. (There were even days when the beloved and experienced music teacher found it difficult to get them to sing.) "Go with the flow."
- Knowing how the children perceive a program such as this is also hard. Attitudes often cannot be seen, and one must be careful not to make assumptions about observed behaviors.
- There is a strong need for concrete, over-arching goals to tie sessions together. (This is one of the major issues we need to address in our second year.)
- Student assistants must be vetted carefully for their ability to work with children. (While we had no major problems with any of our assistants related to their interactions with the children, it is clear that some were far better at helping the children learn and remain on task than others.)
- There is simply no substitute for working with a quality teacher who has a strong rapport with the children and understands where they're coming from. On numerous occasions our music teacher partner pointed out where some of our assessments and impressions of how the program was going were wrong. For example, one student didn't seem to "engage" with the program, so we assumed that she simply didn't want to be in it. Our partner pointed out that all of this student's friends had dropped out of the program for one reason or another, and that the fact that she was still with us was a strong indicator of her desire to be there. We would never have known this, and we quickly learned to rely on our partner's perceptiveness and trust her instincts.
- Despite all the time and patience it takes to get children to focus on learning in an after-school program and the inevitable ups and downs of such an endeavor, there are numerous, priceless, unforgettable "ah-ha" moments that make all the effort worthwhile.

5. FORMAL EVALUATION

Our project was supported by an evaluation team centered in the UMass Lowell Center for Program Evaluation. The team observed after-school sessions, administered surveys to both students and faculty, and conducted focus groups with both students and faculty. The following statements are adapted from the Conclusion section of their Year 1 Evaluation Report:

> Faculty suggested that while end-of-year student programming was not as advanced as they had hoped, it clearly improved during the year. Overall, the students had more exposure to music than computer programming prior to starting the project. Therefore, the learning curve was steeper for computer programming. There were understandably individual differences in interest and skill levels; at least one student exceled in computer programming and was able to assist other students while pushing herself and the facilitators so that she could go further in her programming.
>
> The students' self-efficacy toward teaching someone else to program increased substantially over the course of the project. They seemed to like programming more at the end of the school year and reported an increase in interests in both music and computing programs outside of school.
>
> Students clearly enjoyed sharing their work with other students and the facilitators in the program. They showed confidence in their abilities to create the music on the computers and enjoyed the opportunity to display these abilities.
>
> The majority of the students were from underrepresented groups in STEM. While a shortcoming of the project may be the lack of facilitators from underrepresented groups, the ratio improved over time. Additionally, despite the lack of facilitator role models who "looked like" the student participants, the facilitators were able to connect well with the students.
>
> Most of the data pointed to the primary strength of the project being the relationships that were built between the students and facilitators that allowed the students to feel that they were in a safe place to learn and ask questions. Both the students and facilitators enjoyed working with each other.
>
> It is likely that this opportunity for middle school students, with more one-on-one interaction with students and faculty from the university, will a have long-lasting impact on their views toward education in general and, more specifically, toward the music and computer science fields.

6. ANTICIPATED CHANGES FOR YEAR 2

There are several changes that we plan to make to the program to improve it in its second year:

- We plan to recruit additional student assistants to increase the ratio of facilitators to students. We also hope to recruit more female and minority university students with whom the middle school students can identify. As discussed in Section 4.2, while the middle school students warmed to the white professors and university students over time, we observed that the girls had a stronger connection to our lone female student facilitator.
- We plan to buy new computers that will allow us to have more control over the school computing environment.
- We will attempt to produce a holiday album of both singing and computer-generated music that the students can share with their families and friends to further motivate their involvement in the program.
- We will attempt to have the students build a website that showcases their work and further motivates their involvement in the program.
- We will explore the possibility of partnering with another school also to further motivate involvement in the program.
- We will change some of the members of our advisory board to hopefully improve the feedback we are getting from that group.

We also intend to make better use of possible collaborations within the school itself. For example, the computer teacher has been on our advisory board, but we hope to tap her expertise more directly to gain insights on working with our students. The school also has an award-winning Social Studies teacher who is a technology wiz and who has done collaborative work not only with the students in her own classroom, but also between her students and students in other schools. Both of these teachers may be able to help us with classroom management and possible activity development.

But perhaps most importantly, we intend to expand the "play" and/or "gaming" aspects of the one-on-one and social interactions of the program to improve the quality of teaching and student engagement. Students expressed the desire for such an approach in the focus groups conducted by our evaluation team, and we believe that incorporating a play- or game-based approach will give the students more chances to experiment, work collaboratively, and drive their own learning.

7. ACKNOWLEDGMENTS

This project is supported by Award No. 1515767 from the National Science Foundation Division of Research on Learning. Any opinions, findings, conclusions, or recommendations expressed in this paper are solely those of the authors and do not necessarily reflect the views of the National Science Foundation.

The authors acknowledge the contributions of the following people and thank them for their support of the project, particularly those at the Bartlett Community Partnership School (BCPS):

- Peter Holtz, Principal, BCPS
- Kara Haas, computing teacher, BCPS
- Kristin Weigold, after-school coordinator, BCPS
- Joe Meli, student asst., UMass Lowell CS major
- Jonathan Roche, student asst., UMass Lowell CS major
- Nicole Vasconcelos, student asst., UMass Lowell Music major
- Ben Miller, student asst., UMass Lowell Music major
- David Cherepov, student asst., UMass Lowell Math major
- Andrea Andzenge, evaluator, UMass Lowell Grad School of Ed
- Dan Washington, music arranger, Xerox Corporation and Barbershop Harmony Society

Note: The parents or guardians of all students pictured in this paper have signed IRB-approved photo release forms.

8. REFERENCES

[1] Audacity Open Source Development Team (2011). *Audacity: The Free, Cross-Platform Audio Editor and Recorder*. audacity.sourceforge.net *accessed* Oct. 25, 2014.

[2] Chambers, J. (2016). *An ABC Primer*. trillian.mit.edu/~jc/music/abc/doc/ABCprimer.html *accessed* Aug. 16, 2016.

[3] Delpit, L. (2006). *Other People's Children: Cultural Conflict in the Classroom*. New York: The New Press.

[4] Geiger, T., Parker, D., & Kyriakides, D. (2015). *Stitches*: Hal Leonard Music Publishing.

[5] Greher, G.R., & Heines, J.M. (2009). *Sound Thinking: Conceptualizing the Art and Science of Digital Audio for an Interdisciplinary General Education Course*. Assoc. for Technology in Music Instruction (ATMI) 2009 Conf. Portland, OR.

[6] Hatasuhut, R. (2016). *Web Music Score*. www.ronaldhutasuhut.com/wp-content/wms/WebMusicScore.html *accessed* Aug. 18, 2016.

[7] Heines, J.M., Jeffers, J., & Kuhn, S. (2008). *Performamatics: Experiences with Connecting a Computer Science Course to a Design Arts Course.* Int'l. Jrnl. of Learning **15**(2):9-16.

[8] Heines, J.M., Greher, G.R., & Kuhn, S. (2009). *Music Performamatics: Interdisciplinary Interaction*. Proc. of the 40th ACM Tech. Symposium on CS Education, pp. 478-482. Chattanooga, TN: ACM.

[9] Heines, J.M., Greher, G.R., Ruthmann, S.A., & Reilly, B. (2011). *Two Approaches to Interdisciplinary Computing+ Music Courses.* IEEE Computer **44**(12):25-32.

[10] Ho, W.-C. (2004). *Use of information technology and music learning in the search for quality education.* British Jrnl. of Educational Technology **35**(1):57-67.

[11] Kohl, H. (1994). *"I Won't Learn from You" and Other Thoughts on Creative Maladjustment*. NY: The New Press.

[12] Magerko, B., Freeman, J., McKlin, T., McCoid, S., Jenkins, T., & Livingston, E. (2013). *Tackling engagement in computing with computational music remixing*. Proc. of the 44th ACM Tech. Symposium on CS Education, pp. 657-662.

[13] Martin, F., Greher, G.R., Heines, J.M., Jeffers, J., Kim, H.-J., Kuhn, S., Roehr, K., Selleck, N., Silka, L., & Yanco, H. (2009). *Joining Computing and the Arts at a Mid-Size University.* Jrnl. of Computing Sciences in Colleges **24**(6):87-94.

[14] National Science Foundation (2007). *NSF Award #0722161 - CPATH CB: Performamatics: Connecting Computer Science to the Performing, Fine, and Design Arts*. CNS: Division of Computer and Network Systems, www.nsf.gov/awardsearch/showAward?AWD_ID=0722161 *accessed* Nov. 4, 2013.

[15] National Science Foundation (2011). *NSF Award #1118435 - Computational Thinking through Computing and Music*. DUE: Division of Undergrad. Ed., www.nsf.gov/awardsearch/ showAward?AWD_ID=1118435 *accessed* Nov. 4, 2013.

[16] Pencil Code (2016). *Dream it. Code it*. pencilcode.net *accessed* 8/16/2016.

[17] Platten, R., & Bassett, D. (2015). *Fight Song*: Columbia Records.

[18] Ruthmann, S.A., & Heines, J.M. (2009). *Designing Music Composing Software with and for Middle School Students: A Collaborative Project among Senior Computer Science and Music Education Majors*. Association for Technology in Music Instruction (ATMI) 2009 Conference. Portland, OR.

[19] Ruthmann, S.A., Greher, G.R., & Heines, J.M. (2012). *Real World Projects for Developing Musical and Computational Thinking*. 30th Int'l Society for Music Education (ISME) World Conf. on Music Ed. Thessaloniki, Greece.

[20] Scratch (2016). *Create stories, games, and animations; Share with others around the world*. scratch.mit.edu *accessed* 8/16/2016.

[21] Stinson, L. (2013). *Google and apple alums invent adorable robots that teach kids to code*. *Wired*.

[22] Studer, K. (2005). *Maximum Technology in the Music Classroom: Minimum Requirements.* Teaching Music **13**(3):44-47.

[23] Swift, T., Martin, M., & Shellback (2014). *Shake It Off*: Big Machine Records.

[24] Tatun, B. (1997). *"Why Are All the Black Kids Sitting Together in the Cafeteria" and Other Conversations About Race*. New York: Basic Books.

[25] Trueman, D. (2011, as quoted by Jacqui Cheng). *Musicians, Tune Your Keyboards: Playing in a Laptop Orchestra*. arstechnica.com/gadgets/news/2011/07/laptop-orchestras-what-are-they-and-where-did-they-come-from.ars *accessed* Nov. 14, 2011.

[26] Walshaw, C. (2016). *ABC Notation*. abcnotation.com *accessed* Aug. 16, 2016.