Using Black Music as a bridge to understanding introductory programming concepts

David James
Department of Computer and Information Sciences
Spelman College
Atlanta,GA
djames18@spelman.edu

Lelia Hampton
Department of Computer and Information Sciences
Spelman College
Atlanta,GA
lhampto4@scmail.spelman.edu

***Abstract*— Computer Science is a field that still has long strides to make before its workforce reflects the racial, ethnic and gender diversity found in the United States. To address this problem the authors developed culturally relevant class activities and homework problems that teach students to use loops, lists, functions and conditional logic, by remixing popular Black music in an introductory Python course. The activities were developed with the thought that students would be less intimidated by the content, if they saw their culture reflected in the course materials. Using an online survey, students were asked to compare their experience working on the problems that used Black music as a context, with their experience completing textbook problems on the same content. All of the survey respondents were Black female undergraduate students enrolled in a STEM major at a historically black college.**

Students reported that they were better able to identify with the examples and problems using Black Music as a context. Study participants also indicated that making enjoyable music motivated them to work harder than they would on textbook problems. Of the 33 study participants, 60% had no prior experience programming, and 63% of the students expressed an interest in taking further computer science coursework as an elective. The authors explain the sequence of assignments they used to help students understand introductory computer science concepts, and share lessons learned from using Black Music as a tool in the classroom.

Keywords—STEAM, CSxMusic, culturally responsive pedagogy

I. Introduction

There is a persistent lack of participation in computing, but especially among women, people of color, and women of color. In 2018, only 4% of Bachelor's degrees in computer science were awarded to black women [1]. In order to address participation and interest in computing, music has been used as a novel approach in computer science education to increase engagement and enhance persistence in computing. Most studies occur in K-12 settings [2][3][4][5], or out-of-school settings such as workshops or camps [6]. While some studies include participants higher education [7][8][9], few studies have taken place in higher education environments serving a population of African American women. Our study targets African American women in higher education through the use of hip hop music as a tool for culturally responsive pedagogy in computer science education. To address this gap in the literature we pose the following research question: How does remixing Black Music impact a student's learning experience in an introductory programming course?

II. Methods

The study takes place in an introductory computer science course for non-majors at a four year historically black college. All of the study participants are Black Women who are STEM Majors at the college. The authors drew a pool of participants for the study by taking a convenience sample of students from two of the 5 sections of the introductory Python programming course. 33 of the 53 students enrolled in the two sections of the course responded to the surveys. The authors distributed an online survey at the end of the course, which asked students to compare the music-based problems they'd worked on with textbook problems involving looping and conditional logic. Items on the survey also asked how much experience they've had with programming, and if they had formal training with music before the course.

The authors used a mix of inductive and deductive coding to categorize responses to open ended questions in the survey. The first two themes, relating prior knowledge to new knowledge, and using culture to draw students to the material (connecting abstractions with lived realities) were drawn from Gay's[10] concept of a pedagogical bridge, within her theory of culturally responsive teaching. The third theme (computing as a tool to increase creativity and fun) sprang directly from the students reflections on their experience in the class.

III. Procedure

The remix project was broken into four activities for the students: (1) We introduced study participants to the concept that black music can be constructed using algorithms, by

978-1-7281-7172-2/20/$31.00 ©2020 IEEE

walking them through an algorithm they could use to re-create a popular song. (2) The students were asked to create their own algorithm for a remix by sketching their plan to rearrange premade pieces of a popular song (called samples). (3) After students learned about for and while loops, we taught them how to place their samples in loops to code the remix algorithm they created. (4) The students chose their own song they would remix in two different ways, and used conditional logic to select which remix version they wanted to hear. The students were also given assignments to learn about looping and conditional logic straight from the text book, such as summing up odd or even numbers between a starting and ending point.

A. *Making Algorithms to remix Black Music*

Early in the course the instructor planted the idea that algorithms can be used to create Black Music. In the first class session is dedicated to recreating the song Otis by using an algorithm to rearrange parts of the song Try A Little Tenderness. After the lesson, the students are asked to make a remix of a popular song by flowcharting how its parts should be rearranged. The students were given a PowerPoint slide with buttons that triggered 8 samples of a popular song (Location by Khalid). The participants had to diagram the order of the samples in their remix and how many times each of the samples should be looped. The students were required to have at least four samples and at least two loops in their remixes. During the exercise are instructed to test and revise their remix algorithms. To test their remixes students had to execute their algorithm by hand, by clicking on the samples to play them in the order prescribed by the algorithm. If the remix didn't sound "good", they were asked to edit the algorithm and walk through it until the steps produced a desirable outcome.

B. *Using Python to code the Remix Algorithm*

After teaching the students how to use for and while loops they are introduced to the jythonMusic library [11].The study participants would use this library to play samples and loop them using the python programming language. The IDE included with the library, called JEM (Java Environment for Music) is designed to run jython code, which is python syntax (version 2.7) that can be used to leverage objects and methods written in java libraries. Using sample code the students learn how to store a sample (stored in a .wav music file) into an AudioSample object. Once this object is constructed its methods can be used to play, pause, stop, speed up and slow down a sample, which will be audible through the sound card/speakers of the student's computer.

C. *Remixing their own Music*

Once students became comfortable using the jythonMusic library, they were given an assignment to make at least two remixes of one song that they choose. The remix project required students to create a program that asks a user to choose a version of the remix they would like to hear (via the console window), play the appropriate remix until it ends, and return to ask if they'd like to hear another remix or quit. Each version of the remix they created had to be housed in a function. Before arranging their remixes, the participants had to be taught how to identify a sample, and determine whether that sample sounds good when placed in a loop. Audacity has a few tools (like loop play and the selection tool) that allow students to preview and loop their sample, before its saved to a file.

The students were prompted to identify at least two songs that they would like to work with and emphasized that they should use instrumental versions when possible. Instrumental versions of songs are easier to remix because samples of music without words are easier to loop. Some songs also may not have enough variation to sound like a new work once its samples are rearranged. Since the students will create at least two remixes from one song, there has to be a fair amount of variety with respect to the number of sounds in that song. The instructor advised the students to use song they've chosen as their sole source of samples for the remix. Blending two or more songs together requires the students to find songs that have compatible tempos, keys and rhythms, which is beyond the scope of an introductory computer science course. After completing the assignment students were asked for their thoughts on the music-based activities versus problems that required them to use loops in the textbook.

IV. Findings

In discussing the delivery of instruction to ethnically diverse students, Gay[10] emphasizes the importance of establishing cultural congruity in your instruction. Gay says you accomplish this by spending a great deal of time demonstrating through examples, stories and problems how course concepts operate in practice, through the experience and cultural lenses of your students. She calls the use of these culturally relevant examples "pedagogical bridges to connect prior knowledge to new knowledge, the known with the unknown, and abstractions with realities.", and highlights the positive impact it has on academic achievement.

In our survey students were asked to compare textbook problems that used loops (such as counting and summing even and odd numbers) to their remix project, which asks them to use for and while loops to rearrange a song they choose. The students were given about two weeks to complete the project. The authors also asked students whether they would like to see Black music used more or less frequently as a context for computer science instruction, and why they chose to answer the way that they did. Students were asked to state their major, number of years of experience with music, years of experience with programming and whether they would like to take further computer science course work. Most of the participants in the study had no prior programming experience and no previous training in music.

Although this class is the only computer science course required by their majors, over 60 percent of the study participants showed an interest in taking future computer science coursework. It's quite encouraging to see that students did not have to have a musical background to learn programming concepts through remixing. As an instructor, I did have to emphasize that students wouldn't be graded on the quality of the music they produced. Below in Table 1 the authors have provided a summary of responses from the Likert scale items in the web survey.

TABLE I. SUMMARY OF SURVEY RESPONSES

Programming Experience	Prior Music Training	Desire to take more classes in computing
20/33(60.6%) of students had no experience	21/33(63.6%) of students had no training	21/33 (63.6%) of students would certainly take more courses
10/33(29%) had less than one year of experience	2/33 (6.06%) had less than a year of training	5/33 said they may take more courses
Three students (9%) had more than one year of experience	9/33(27.27%) had 3+ years of training	7/33 students will not take more courses

We will present the findings from the open ended questions on the survey in three themes. Two of the themes are derived from Gay's concept of a pedagogical bridge, and one inductive theme sprang from the students responses. The two themes from Gay's work[10] are relating prior knowledge to new the use of culture as a magnet to the course material. The third theme which falls outside of the concept of the pedagogical bridge is called computing as a tool to increase creativity and fun.

A. *Theme 1:Pedagogical Bridges - Relating prior knowledge to new knowledge*

One finding we have discovered that is in line with Gay's idea of a pedagogical bridge is that the music helped students to better understand programming concepts. Understanding how to use loops, and trace the flow of their program, made it easier for them to understand how to modify their code for a better outcome. Students gave the following responses when asked if they would like to see more problems in the future that use Black music as a context.

> "I would like to use more problems that use black music as context in computer science because I identify as Black and I enjoy music. I think it is a great learning tool to use music that (I like) to create a remix. I do not see a purpose in using music or other tools that I cannot relate to from personal experience. We learn better when we use concepts and tools that WE enjoy using (for instance black music was used to help black students learn a new topic of interest)."

> "I like the idea of using music because it is something that is more personal and easy to connect to than just a prompt saying to add numbers or do a task. It allows for creativity and motivates me to actually understand what i am doing so that i get a product i like and am proud of.."

B. *Theme 2: The use of culture to draw students to the course material (connecting abstractions with lived realities)*

Another helpful finding of the study is that our students are constantly connected to Black Music, through Spotify, Apple Music, YouTube and traditional radio. Since music is a part of their daily experience, they are already aware of the sound of a good remix. Students worked on their project with a picture of the polished sound that they'd often heard, and it pushed them to work on the project longer than they would have, if the project did not have a context that was familiar and important to them. Students submitted the following responses when asked whether they cared about the quality of the remix, and if that impacted the amount of time spent on the project.

> "Yes, when listening to song remixes on the radio they are still smooth and seamless. So, when completing my project I want to have that same outcome. Wanting to have that specific outcome did make the time spent on the project even longer."

> "Well the remix had to sound right, so that meant that I had to put in more time, opposed to me just adding random numbers together. (comparing to problems from the text)"

> "In order to get my remix to flow like a natural song, I worked for hours to ensure that it was work that I was proud of."

Using Black music as a cultural tie to the material in this class also prompted students to share their code and remixes with friends, due to their familiarity and love of black music.

> "I would like more black music because it takes computer science outside of basic computing and makes it fun. This project is something I was able to show my producer uncle and he thought it was so cool"

> "One of my favorite parts of this class was the fact that it involved such a strong music base. I was kind of scared when starting this class because I have no previous computer science background but the fact that it contained such a strong music base made me really fall in love with it. I told some of my friends about the process we were going through and how excited I was about it and they thought it was interesting as well."

> "I think there should be more use of black music in the context of a computer science class because black music uses different concepts of computer science (remixes, loops, lists, etc.) I thought that it was interesting that black music was so relatable and a great example of how even though computer science is a largely white field and industry, the concepts of computer science can all be found in many different genres of black music."

The infusion of black music into the course made the material more approachable, by helping students to see themselves reflected in the material. This was especially important since the course is one that is taken largely by non computing majors, many of which (48%) wait until they are upperclassmen to enroll. Some participants said that they perceive computer science to be an intimidating discipline dominated by people who don't look like them.

C. *Theme 3: Computing as a tool to increase creativity and fun*

Many of the students also remarked that they had fun working on this project, and that it helped them to be creative in ways that they hadn't imagined before enrolling in the class.

Some of the students also expanded their view of what computer scientists do as a result.

> "The remix project was one of my favorite projects because I got to choose a song that I really like and work with it to create something. Although it was challenging at times because some beats didn't sound right or make a smooth transitions, I found that was the part I liked the most. I got to be creative and I understood that it was ok if it did not sound perfect."

> "I really enjoyed the music remix project. It gave me a chance to be creative and unique while simultaneously learning about programming."

> "The project was more interesting to me. I was engaged because it dealt with something that involved music. I love listening to music so it was fun learning how to create my own remix to my favorite song and write a code to allow it to play. The (textbook) assignments that we have done do not engage me as much. They seem to be …irrelevant to my life."

V. Lessons Learned

Culturally responsive theories of pedagogy bring home the importance of understanding students' culture, and underscores the value of expertise that students already have. Within the context of the classroom, it's important to understand the current music your students like and weave that culture throughout examples in your courses. Students know exactly what music excites them, so you can ask what their favorite song and artists are at the beginning of the semester, and create some samples from those songs. Also, take the time to teach students how to make samples from their own favorite songs, which allows students to work with the most current songs that move them. At the start of this process it is helpful to have samples that are already cut and placed in the right directories for sample code to work. This cuts out pesky errors at the outset when students are just starting out and need quick results to stay motivated. I have had students lose interest, after getting repeated errors from source code or music samples being placed in the wrong directory.

It's also important to emphasize to students that they are not being judged by the quality of their music production, but they are being judged on how well they are able to understand how to use conditional logic, loops and functions. Some students in the study got so caught up tweaking the sounds that they forgot to finish a working prototype that satisfies the requirements.

References

[1] Computing Research Association, "CRA Taulbee Survey," Comput. Ressearch News, vol. 31, no. 5, 2019.

[2] R. Eglash, M. Krishnamoorthy, J. Sanchez, and A. Woodbridge, "Fractal Simulations of African Design in Pre-College Computing Education," Trans. Comput. Educ., vol. 11, no. 3, pp. 17:1--17:14, Oct. 2011.

[3] J. Freeman et al., "Engaging Underrepresented Groups in High School Introductory Computing Through Computational Remixing with EarSketch," in Proceedings of the 45th ACM Technical Symposium on Computer Science Education, 2014, pp. 85–90.

[4] R. B. Shapiro, A. Kelly, M. Ahrens, B. Johnson, H. Politi, and R. Fiebrink, "Tangible Distributed Computer Music for Youth," Comput. Music J., vol. 41, no. 2, pp. 52–68, May 2017.

[5] A. L. Meyers, M. C. Cole, E. Korth, and S. Pluta, "Musicomputation: Teaching Computer Science to Teenage Musicians," in Proceedings of the Seventh ACM Conference on Creativity and Cognition, 2009, pp. 29–38.

[6] Y. Kafai, K. Searle, C. Martinez, and B. Brayboy, "Ethnocomputing with Electronic Textiles: Culturally Responsive Open Design to Broaden Participation in Computing in American Indian Youth and Communities," in Proceedings of the 45th ACM Technical Symposium on Computer Science Education, 2014, pp. 241–246.

[7] S. Siva, T. Im, T. McKlin, J. Freeman, and B. Magerko, "Using Music to Engage Students in an Introductory Undergraduate Programming Course for Non-Majors," in Proceedings of the 49th ACM Technical Symposium on Computer Science Education, 2018, pp. 975–980.

[8] A. Misra, D. Blank, and D. Kumar, "A Music Context for Teaching Introductory Computing," in Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education, 2009, pp. 248–252.

[9] D. Fields, V. Vasudevan, and Y. B. Kafai, "The programmers' collective: fostering participatory culture by making music videos in a high school Scratch coding workshop," Interact. Learn. Environ., vol. 23, no. 5, pp. 613–633, 2015.

[10] G. Gay, "Preparing for Culturally Responsive Teaching," J. Teach. Educ., vol. 53, no. 2, pp. 106–116, 2002.

[11] B. Manaris and A. Brown, "Making Music with Computers |Creative Programming in Python." [Online]. Available: https://jythonmusic.me. [Accessed: 10-Jan-2019].