Broadening Computing Participation in the Navajo Nation

Ashish Amresh
ashish.amresh@nau.edu
Northern Arizona University
Flagstaff, AZ, United States

Jeffrey Hovermill
Jeffrey.hovermill@nau.edu
Northern Arizona University
Flagstaff,AZ, United States

Wei Yan
wei.yan@nau.edu
Northern Arizona University
Flagstaff,AZ, United States

Paige Prescott
paige@computersciencealliance.org
Computer Science Alliance
Santa Fe,NM, United States

Figure 1: Sample Apps developed by teachers

ABSTRACT

Native Americans (NA) have historically been the most underrepresented population when it comes to participating in STEM and computing careers. The Navajo are one of the country's largest NA groups, and understanding the barriers and developing solutions to increase their participation will have far-reaching consequences on informing the research and practice on how computing can be taught at NA-serving high schools. The paper describes the experience gained over three years of working in this region via project *Let's Talk Code*, which aims to help math, science, and art teachers from Navajo high schools develop CS-based projects in their existing courses and provide mentorship and guidance. *Let's Talk Code* is constructed as a research-practice partnership (RPP) where the teachers (practitioners) work with a multi-institutional team of researchers and CS educators to improve the capacity-building needs of its partners (high schools). The paper details the evolution of the project over the years and highlights challenges, barriers, and strategies that were used to impact a significant number of teachers throughout the project.

ITiCSE 2024, July 8–10, 2024, Milan, Italy

ACM ISBN 979-8-4007-0600-4/24/07
https://doi.org/10.1145/3649217.3653551

CCS CONCEPTS

• Applied computing → Education; **• Social and professional topics → K-12 education**; **Computer science education**; **Computational thinking**.

KEYWORDS

Computing, Broadening Participation, Research Practitioner Partnership, Indigenous Communities

ACM Reference Format:
Ashish Amresh, Jeffrey Hovermill, Wei Yan, and Paige Prescott. 2024. Broadening Computing Participation in the Navajo Nation. In *Proceedings of the 2024 Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2024), July 8–10, 2024, Milan, Italy.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3649217.3653551

1 INTRODUCTION

Computer Science (CS) education has traditionally adopted a content-agnostic and concept-centering approach to developing instructional strategies for K-12 [10, 22]. While these strategies have led to some academic success among majority-white schools, this approach has been less effective among underrepresented minority groups, such as Native Americans (NA). Most programs that have emerged from federal and foundational grants to improve CS education across the K-12 system have not addressed the specific concerns, context, and needs of Native American (NA)-serving schools [7]. The widespread need for more diversity and representation in the STEM workforce, specifically among Native Americans, further supports the need to provide NA students access to rigorous and relevant CS education that addresses their contextual needs.

Computing has the lowest representation of NA students among all STEM fields [9]. The underrepresentation of NA students in computing is a critical concern, especially with the United States facing a shortage of computer scientists. In 1997, 1.3% of the undergraduate Native American population earned a CS degree compared to 2% of the overall population. Since then, there has been a significant growth in CS degrees earned by the general population, almost doubling to close to 4% of the degrees awarded in 2016, yet the numbers for CS NA degrees have declined to 1% of their population. Research shows that NA students thrive best when the curriculum connects to their culture [3, 12, 15]. Culturally connected curricula, such as the Smithsonian's National Museum's Native Knowledge 360 [20], have shown increased outcomes for NA-serving schools. Education stakeholders, including local Tribal Colleges and regional Universities, have individually worked to increase the broadening of CS education for NA students living on or around the Navajo Reservation. While these efforts have demonstrated limited success, they are often time-bound to a specific funded initiative and lack adequate research and evaluation to demonstrate potential efficacy as a long-term strategy. There is a strong need to combine the resources, expertise, and cultural/traditional knowledge of education stakeholders serving Native American students to identify the most effective and sustainable approach to increasing CS access, interest, and college and career success for NA high school students, living on or near the Navajo Nation. This paper provides a report of experiences, findings, and strategies undertaken by Project *Let's Talk Code* to address the issue of broadening participation in computing in and around the Navajo Nation.

The major goals for the project were:

(1) Develop and assess the effectiveness of a culturally connected approach to CS education for NA students.
(2) Increase teacher confidence and competence to introduce CS concepts to NA students.
(3) Increase computing awareness and interest among students in and around the Navajo Nation.

After running the project *Let's Talk Code* for three years from 2021 - 2023, we report on the strategies and outcomes for each of the above goals for the project. We provide the background information (Sec. 2), explain our target population demographics and the methods used with the teachers via different professional development activities (Sec. 3), show sample lesson plans developed by their teachers (Sec. 4), and highlight our observations, outcomes and future directions (Sec. 5).

2 BACKGROUND

2.1 Prior Knowledge

Programming can be scary and confusing for novices [13]. One of the biggest challenges in addressing the need for broadening participation is finding an efficient way to introduce programming to beginners in K-12 settings. The academic community has responded by using simplified languages and simplified ways to write programs. When introducing programming to students, much emphasis has been on hiding the intricacies of the program execution environment and the interaction of hardware and software. This approach, which typically includes graphical programming interfaces, reduces the complexity that a beginner programmer needs to handle. The introduction of the Advanced Placement(AP) Computer Science Principles(CSP) course and several pilot studies funded by the NSF [2] reveal the widespread use and adoption of intuitive graphical programming interfaces and their success in improving CS outcomes at high schools.

Evidence of such efforts with the NA populations is seen in a recently completed effort by the Arizona Governor's office, Intel, and the Science Foundation of Arizona via the Native American Code Writers Program (NACWP) [18]. Pilot efforts carried out by this program have shown limited success in building teacher capacities among the Gila and Yavapai tribes by conducting professional development workshops using the AP CSP curriculum framework provided by Code.org [1, 4]. One of the observations of the pilot program was that the curriculum did not seem to address the needs of the target population and seemed disconnected from most students. A significant observation by Dr. Linda Coyle, one of the advisors, was that students did not seem to sustain the interest as it became too foreign or disconnected, especially when advanced programming modules that require iteration or recursion were taught. To address these gaps and further improve the use of the Code.org framework, we engaged teachers to introduce computing in their current courses by building lessons that would be strongly connected to the target population.

2.2 Situated Learning

Helping students build cognitive connections between coding and visualizing its output is only part of the solution; it's a common theme in teacher preparation and professional development efforts. Simply writing code without real purpose or impact on real-world problems can demotivate all students, particularly female students [14]. Creating contexts for learning where the tasks are performed and contributing to solutions to real-world problems can bolster learning and student engagement [19]. Situated learning theory states that learning happens best when conducted in the environment in which it is used [6, 16]. Situated learning theorists posit that all learning is situated in specific contexts and that the contexts in which people learn can help or hinder their learning. A detailed review provides input into the success of such strategies for indigenous youth[8], and more recently, it has been shown that contextualizing culture in a digital/CS learning program that designed textiles helped improve the identity gap in NA teenage girls [24]. Thus, an approach that supports novices, particularly NA students, to move beyond the introductory stage of learning toward more profound engagement with complex programming is to provide situated and engaging tasks tied to specific coding elements and use those elements to solve real-world problems. These situated activities can bolster engagement and learning by connecting the 'sandbox' exploration of how coding impacts and interacts with NA culture to concrete problem-solving drawn from realistic, real-world examples. Code.org currently does not have these tailored structures built into its CS educational framework, and this project makes exploratory advances to add cultural and community-based connectivity to the framework.

2.3 Contextualizing Computing

Guidelines for engaging NA students in STEM [25] recommends that connecting education to relevant community needs and cultures is paramount to their success. In this project, we use contextualized instructional lessons, where teachers build Apps during the CS instruction as their future code.org curricular units to teach NA students basic programming, and this approach has been successful in previous research [15]. In the NACWP final report, the top three recommendations to boost NA's participation in STEM and computing are as follows:

(1) Improve access to curriculum, technology, and internet.
(2) Delineate an early and clear pathway to success.
(3) Align cultural identity with STEM necessity.

Using situated and context-sensitive scenario-based problems, we aim to contribute to recommendations (1) and (3). Students are introduced to programming using real-world scenarios and work towards finding a solution while learning about the interaction between software and hardware. The project tests various real-world problem-based modules and assesses their impact. To address (2), we develop a series of best practices by iterating through several such scenarios, improving their application and effect. These activities demonstrate success and remove barriers related to inaccurate stereotypes, such as low confidence in computing skills. We expect this setting, along with mentoring (project team) and role models (native undergraduate CS students at Northern Arizona University), can break the stereotype, provide a positive learning environment, and improve students' perception of their capabilities in programming. In a prior pilot effort that was undertaken by the Girl Scouts of Arizona and educated over 300 middle and high school girls [23], including those from the NA tribal areas by teaching introductory programming principles in a gaming context, it was observed that the students who had the highest initial interest were those from the tribal regions. However, their interest could not be sustained during the program, and they had the lowest completion rate.

3 METHOD

3.1 Target Population

Before teachers can be recruited, IRB approval is needed at the project's institution and by the Navajo Nation Human Research Review Board (NNHRRB). This includes getting approval letters from the school principal, the school superintendent, and the school board. The process has to be followed for each participating school in the Navajo Nation. Additional approval from the Navajo Department of Dine Education (NODE) for the project's activities is required. The NNHRRB will then invite investigators to present their projects at a monthly board meeting and vote on approving the activities. Due to the nature of what is allowed by the board, the project took a simple approach to data collection using teacher and student surveys (Sec. 5) for reporting the outcomes. The project targeted teachers from the schools listed in Table 1 and worked with over 35 teachers (in science, mathematics, and art) across the schools. The student population at these schools has high rates of free and reduced lunch and Math and Science achievement rates that are lower than the state average. Early on, the professional learning for the teachers was handled online (due to the COVID-19 pandemic). In later years, it moved to in-person summer workshops and cohorts at specific schools during the academic year. The project evolved over the years and worked with teachers in multiple models. The first groups of teachers participated via synchronous online professional activities. The sessions were held in Zoom and breakout rooms with small groups of teachers were created to provide mentoring and guidance needed to complete the CS principles lessons in the Code.org curricular framework. While the learning activities were successful in participation and completion, lesson implementation in the classroom and knowledge transferring among students were limited. The project changed the original approach of building the professional capacity of teachers and offering computing courses in their schools to having teachers introduce computing to their students via their current courses. This enabled enormous interest and participation as the burden of developing and offering CS courses was removed. In this new approach, the team recruited teachers into two different models of professional learning: (1) a professional development summer workshop where teachers from multiple schools would attend at one location and (2) a cohort model for professional development where the project team would pick a school/district and offer the sessions via a series of weekend days during the school year. This strategy enabled the team to reach a large number of teachers as it offered the flexibility that traditional professional development programs do not.

Table 1: Participating teachers and school demographics

School Name	Flagstaff	Hopi	Newcomb	Tuba City	Red Mesa	Ganado	Winslow	Pinon
Type	APS	BIE	NMPS	APS	APS	APS	APS	APS
Median Income	51K	18K	37K	39K	28	23K	42K	25K
Percent Native American	14	96	83	93	97	95	40	95
Percent Poverty	8	43	39	24	33	29	30	48
Number of Teachers	5	15	2	2	6	2	1	2

3.2 Professional Development Plan

Our research-practice partnership (RPP) model follows an evidence-based design, implementation, and revision process. In this model, the lesson plans are co-developed with the participating teachers; researchers and experts guide and mentor classroom implementations. This process was iterated for three years to ensure continuous and sustained improvement. The K-12 CS Framework and CSTA K-12 CS Standards[21] recommends increased integration of CS content in K-12 education programs. For successful implementation of CS programs, where technology-enhanced inquiry materials are critical, teachers must be well-prepared. Current teachers are not "up to speed" with the internet; mobile technologies, social interfaces, and other technologies are mainstream [17].

Furthermore, many lack knowledge of the current technology-supported programs in their own and sister fields that could be implemented to enhance student learning [5]. This situation, coupled with the fact that a great many teachers are ill-prepared to mentor student explorations of problems that require the application of concepts, skills, reasoning methods, and representational forms from more than one content area for their solutions, supports the need for innovative professional development programs [11]. The project therefore includes a participatory model for teachers to be included and contribute to the development and testing of the instructional prototypes throughout the project and provides guidance and mentoring for continued teacher success after the completion of the project.

The professional development sessions were built on the CSP course offered by code.org. We revised the original lesson to a four-day PD:

- Day 1. Introduction to CS (digital representation, bits, information systems) changed to Introduction to App Development (pair programming, data types, design mode).
- Day 2. How the Internet works (networks, protocols, impacts on society) changed to App Design Framework (paper prototyping, programming basics, debugging).
- Day 3. Design your App (paper prototyping) changed to Advanced Programming (app screens, interactivity, data, input and output).
- Day 4. Introduction to Programming (design to code) changed to culturally relevant Lesson Plan Development (connect code to student culture, curriculum, and learning goals).

Our early sessions were held online due to COVID-19. We ran a series of sessions that followed the Code.org curricular framework, giving teachers ample time to design via App development activities that would be contextually relevant and interesting for their students. Then, we revised the original sequence of sessions offered during COVID-19 to in-person sessions. We changed our strategy for our in-person sessions in the subsequent years mainly due to the following observations: (1) teachers had a hard time following and connecting all the materials in the CS principles framework to contextualize to their students and their classes, and (2) the teachers were predominantly from non-CS backgrounds and had limited prior knowledge of coding or computing. We changed the sessions to focus on App development and tasked each teacher to design an app that would replace one week of instruction in their current courses. The App had to introduce computing to their students while keeping with the course curriculum (Table 2).

Table 2: Summary of sample teacher-developed lesson plans

Lesson Plan Name	Course	Description	Learning Objectives	Computing Objectives
Picachu Possibilities	Math	Explore various Picachu Cards and translate their properties into a selectable Application	Learn about probability distribution and ratios	Learn about events, mouse clicks, input, output and loops.
Hopi Natoya	Social Studies	Illustrate the various Hopi tribal communities, their ceremonies and mascots	Discover Hopi culture, trace your roots and origins	Navigation, menus, Lists
Radiant Energy	Science	Discuss different forms of energy and their impact on community	Learn a scientific topic conduct investigation	Create a quiz, verify data accuracy
Binary Converter	Math	Illustrate how two different number systems operate	Learn how to convert one number system into another	Data manipulation, arrays and loops
Heev'a	Language	Show how to identify Hopi insects, birds and flowers	Learn about the Hopi community and language	Navigation, branching event management, audio processing

4 SAMPLE LESSON PLANS

Teachers recruited for the project represented multiple disciplines, such as math, science, languages, arts, and social studies. We provided the lesson plan template and relevant teaching instructions so that teachers could focus more on content creation and teaching flows. The goal of the lesson plan development activity was two-fold: 1) create a Code.org App that introduces a curricular concept from their existing course and 2) introduce computing via the App to the students and improve their participation and interest. The project team and its cohort of teacher mentors (comprised of CS undergraduate students) worked with the teachers via working-group sessions and ad-hoc meetings following the professional learning activities to formalize the lesson plan development and prepare it for week-long student instruction. Table 2 summarizes some of the lesson plans developed by the teachers.

4.1 Lesson Plan Deep Dive

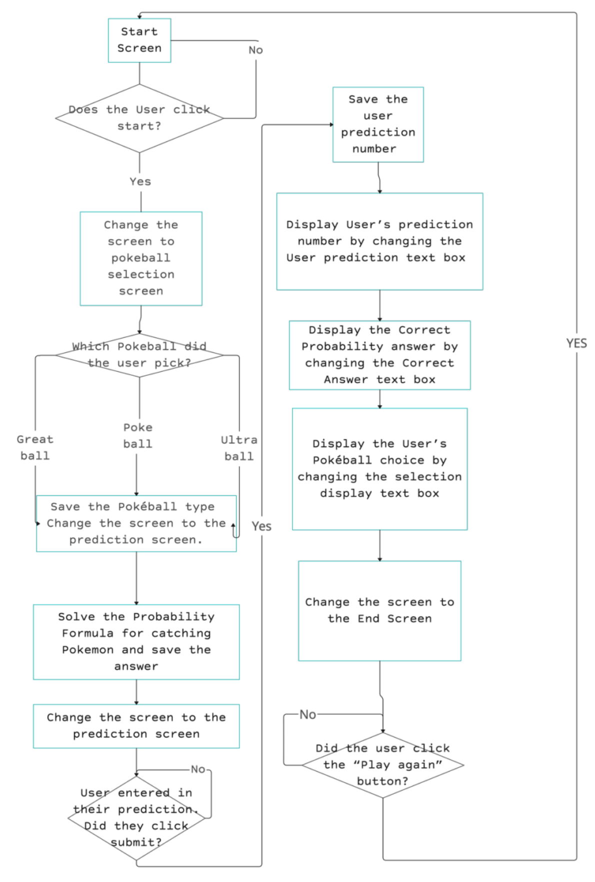

Figure 2: App interaction flowchart

We describe one of our lesson plans developed to help replace a week of instruction covering an introduction to probability and how the teacher used the combination of App development and math instruction to achieve the project's objectives. Table 3 provides a snapshot of the lesson plan's major activities during the classroom implementation (Pokemon characters and Pikachu shown in table 3) are copyrighted and trademarked by The Pokemon Company International). These include creating and setting up Code.org accounts, exploring a sample app that was developed as a starting point for the instruction, investigating and understanding the code and how it ties to the design of the screens within the App, modifying the App by meeting the lesson plan objectives, learning the mathematical foundations of probability and coding those into the App. Figure 2 shows the entire interaction flowchart for the App that students have to develop and can customize based on their liking.

Table 3: Combining coding with math - lesson plan details

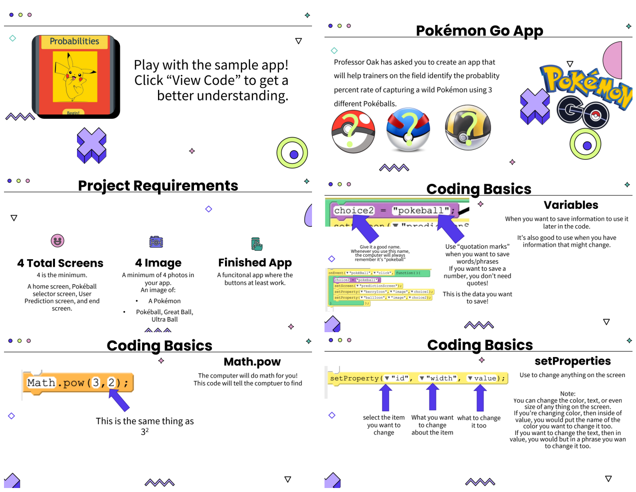

Table 4: Teacher performance in different modalities

Mode of PD	No. of Teachers	No. of Lessons	Mean	Standard Deviation
Online	15	100	74.5	34.8
Summer	8	50	35.5	9.3
Academic Year	12	45	38.6	8.3

5 DISCUSSION

The project has had moderate success in broadening CS participation in remote and rural areas of the Navajo Nation. The major accomplishment has been the ability to get teachers from all disciplines to engage with computing and get excited to introduce computing via the process of creating Apps with their students. While the original expectation was that the majority of the teachers would bring their local community and culture and contextualize the lessons for their students, it was observed that most were not particularly inclined to do so. This could be due to several reasons, such as 1) the teachers worrying that the connection to culture and community might not be attractive to the students and 2) the large proportion of non-native teachers, many of whom being international (due to teacher shortage in the region), which could have created a hesitation on including community and culture.

Students in the Pokemon probabilities class were asked two pre and post-questions: 1) How much they knew about coding (knowledge gain) and 2) How interested they were about coding (interest gain) via a 5-point Likert scale that included 1) Not Confident, 2) Somewhat confident, 3) Confident, 4) Very Confident, 5) Extremely Confident. For a N=37, the knowledge gain jumped from 1.92 to 3.03, and the interest gain jumped from 2.27 to 3.14. Some of the comments that were common among the surveyed students included 1) "I didn't know there was so much to coding", 2) "Enjoyed being able to make an App", 3) "Liked playing with Pokemon" and 4) "Being able to get jobs".

5.1 Teacher Progress and Completion

Code.org provides rich and detailed data on how teachers progress through the lessons during professional training activities. The project embeds all teacher activities and submissions within Code.org, and Table 4 presents the comparison of teacher completion rates between the three professional learning strategies: 1) asynchronous online, 2) summer on-site with multiple schools participating, and 3) academic year on-site at a single school/district. Online modalities experienced high variability in teacher performance, and the in-person ones lowered the variability significantly. The in-person academic year model, where professional learning is spread across multiple weekend days, improved the performance slightly over the summer workshop, where teachers visit a specific location and are provided professional learning over a series of consecutive days. The online sessions covered extensive lessons and were broader in their curriculum delivery. In contrast, the in-person modes were focused on integrating App development activities and building a week-long culturally relevant lesson plan.

5.2 Lessons Learned

The project provided multiple insights into the challenges, strategies, and partnerships witnessed for achieving success with broadening computing in and around the Navajo Nation. We have broken down these into three major categories:

5.2.1 Logistical Observations. It takes enormous patience and persistence to build trust with tribal partners such as review boards, administrative councils, school boards, and districts. Collecting data for research becomes a significant challenge as there is skepticism about research and data collection in rural and tribal areas. The consenting process can be challenging, and protocols and administrative hurdles could limit the ability to disseminate findings. Always allow significant time for approvals as they could take multiple iterations, and scheduling challenges for approvals, such as lack of quorum, are common hurdles. One way to overcome these challenges is to build the logistical issues into project management and have a way to track progress on partnership-building activities.

5.2.2 Curricular Observations. Teachers who represent the Navajo Nation area schools have had very little exposure to technology. Moreover, they have limited time to engage and persist with new skill-building activities due to overburdening efforts. It is therefore of utmost importance to make teachers learning goals and activities explicit. Teachers can easily get into the "follow the instructor" trap, and it may appear that they are completing the lessons without building the necessary expertise. It would be necessary to include ways to bring their creativity and skill into the development of the activities. We achieved this by having teachers work in pairs and alternate who would design an App screen and who would code the design. This helped them work together with their peers and apply the skills throughout the learning sessions. Not all teachers ramp up with the lesson progression at the same pace, and it is therefore necessary to keep track of how they are progressing and on-demand allocate resources such as mentors to help guide them with the progression. Code.org provides a visual interface where teacher progression can be tracked for every level and its set of activities. While there is a strong emphasis on connecting curriculum with culture throughout the broadening participation in CS literature, we observed that many teachers were not always ready to do so. This indicates that a culturally responsive strategy should be applied only after teachers have gained some proficiency with coding to avoid a layer of burden that could come in the way of building their skills and introducing computing to their students. NA-serving schools need ongoing and frequent exposure to coding and learning activities, as evidenced by the in-person academic year model, where the activities were spread out over an academic year, and repeated exposure helped teachers persist better compared to the other modalities.

5.2.3 Research and Dissemination Observations. As a researcher looking to get sound outcomes by following data-gathering practices, it would be essential to understand the challenges and limitations of collecting the necessary data. We recommend an outcomes-focused strategy where building partnerships, iterating with the curriculum, and providing frequent exposure to the participants take precedence over studying and analyzing datasets and drawing conclusions on success indicators. Additionally, we recommend looking at other ways to disseminate findings, such as having teachers present their work at conferences such as the American Indian Science and Engineering Society conference and the Four Corners CS conference, mentoring student teams and having them participate in competitions, and encouraging the teachers to offer out-of-school programs that can build and persist their skills.

6 CONCLUSIONS

This experience report shares our findings with Project *Let's Talk Code*, a three-year effort to broaden CS participation in the Navajo Nation region. Constructed as a research practitioner partnership that aimed to integrate culture in CS lessons and curriculum design and improve CS motivation and interest among teachers and students. The paper describes the evolution of professional development activities over the three years and its impact on teacher and student performance. As we approach the last stage of this project, our next steps include compiling a web-based repository of all the teacher-developed lesson plans and creating an ongoing forum where teachers continue to interact and build their expertise. Our contributions and findings would benefit projects that aim to develop strategies for broadening CS in rural and tribal communities.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation's Computer Science for All program award # 2031505. The work is also reviewed and approved by the Navajo Nation Human Research Review Board (protocol # NNR-22.460).

REFERENCES

[1] AppLab 2024. *Code.org.* Retrieved April 1, 2024 from https://studio.code.org/projects/applab/
[2] Owen Astrachan, Tiffany Barnes, Daniel D Garcia, Jody Paul, Beth Simon, and Larry Snyder. 2011. CS principles: piloting a new course at national scale. In *Proceedings of the 42nd ACM technical symposium on Computer science education.* 397–398.
[3] Network Ray Barnhardt. 2014. Creating a place for indigenous knowledge in education: The Alaska native knowledge network. In *Place-based education in the global age.* Routledge, 137–158.
[4] Rolando Barradas, José Alberto Lencastre, Salviano Soares, and António Valente. 2020. Developing computational thinking in early ages: A review of the Code.org platform. (2020).
[5] Damian Bebell and Laura O'Dwyer. 2010. Educational outcomes and research from 1: 1 computing settings. *The journal of technology, learning and assessment* 9, 1 (2010).
[6] John Seely Brown, Allan Collins, and Paul Duguid. 1989. Situated cognition and the culture of learning. *1989* 18, 1 (1989), 32–42.
[7] Cary Carney. 2017. *Native American higher education in the United States.* Routledge.
[8] Angelina E Castagno and Bryan McKinley Jones Brayboy. 2008. Culturally responsive schooling for Indigenous youth: A review of the literature. *Review of educational research* 78, 4 (2008), 941–993.
[9] Joseph R Cimpian, Taek H Kim, and Zachary T McDermott. 2020. Understanding persistent gender gaps in STEM. *Science* 368, 6497 (2020), 1317–1319.
[10] Leigh Ann DeLyser and Lauren Wright. 2018. Creating a Landscape of K-12 CS Curriculum. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education.* 1098–1098.
[11] Libby F Gerard, Keisha Varma, Stephanie B Corliss, and Marcia C Linn. 2011. Professional development for technology-enhanced inquiry science. *Review of educational research* 81, 3 (2011), 408–448.
[12] Keith James. 2015. Science and Native American Communities. In *Science, Religion and Society.* Routledge, 48–60.
[13] T Jenkins. 2002. On the Difficulty of Learning to Program,[w:] Proceedings for the 3rd Annual conference of the LTSN Centre for Information and Computer Sciences. *Loughborough, UK* (2002).
[14] Tony Jenkins and John Davy. 2000. Dealing with diversity in introductory programming. *Proc. 1st Annu. LTSN-ICS* (2000), 81–87.
[15] Yasmin B Kafai, Eunkyoung Lee, Kristin Searle, Deborah Fields, Eliot Kaplan, and Debora Lui. 2014. A crafts-oriented approach to computing in high school: Introducing computational concepts, practices, and perspectives with electronic textiles. *ACM Transactions on Computing Education (TOCE)* 14, 1 (2014), 1–20.
[16] Jean Lave and Etienne Wenger. 1991. *Situated learning: Legitimate peripheral participation.* Cambridge university press.
[17] Mary Madden, Amanda Lenhart, Maeve Duggan, Sandra Cortesi, and Urs Gasser. 2005. Teens and technology 2013.
[18] NACWP 2019. *Native American Code Writers Program.* Retrieved Jan 21, 2024 from https://www.azed.gov/sites/default/files/2020/10/NACWP%20Final%20Report%202019%20%28002%29.pdf
[19] Brian Nelson, Diane Jass Ketelhut, Jody Clarke, Cassie Bowman, and Chris Dede. 2005. Design-based research strategies for developing a scientific inquiry curriculum in a multi-user virtual environment. *Educational Technology* (2005), 21–28.
[20] NMAI 2022. *National Museum of the American Indian.* Retrieved Jan 21, 2024 from https://americanindian.si.edu/nk360/about/native-knowledge-360
[21] Miranda C Parker and Leigh Ann DeLyser. 2017. Concepts and practices: Designing and developing a modern k-12 cs framework. In *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education.* 453–458.
[22] Lori Pollock, Crystalla Mouza, Amanda Czik, Alexis Little, Debra Coffey, and Joan Buttram. 2017. From professional development to the classroom: Findings from CS K-12 teachers. In *Proceedings of the 2017 acm sigcse technical symposium on computer science education.* 477–482.
[23] scouts 2016. *Girl Scouts Coding.* Retrieved Jan 21, 2024 from https://www.flagstaffstemcity.com/blog/girl-scouts-create-video-games-at-nau
[24] Kristin A Searle and Yasmin B Kafai. 2015. Culturally responsive making with American Indian girls: Bridging the identity gap in crafting and computing with electronic textiles. In *Proceedings of the third conference on genderIT.* 9–16.
[25] Harold L Sorkness and Lynn Kelting-Gibson. 2006. Effective teaching strategies for engaging Native American students. In *National Association of Native American Studies Conference, Baton Rouge, LA.*