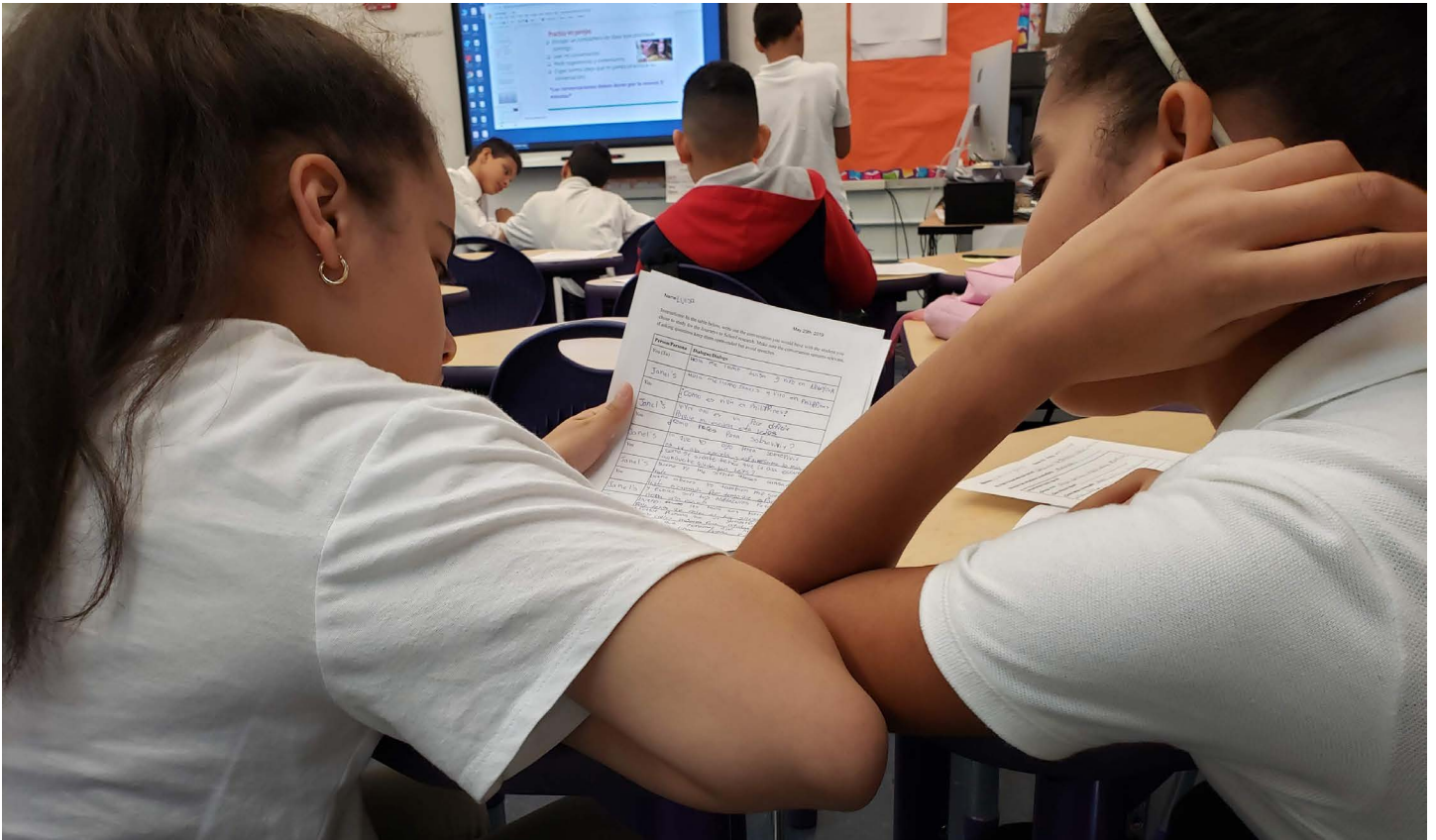


# Our Approach



Our research practice partnership, **Participating in Literacies and Computer Science (PiLa-CS)** promotes equity in computer science education (CS Ed) for emergent bilingual or multilingual learners (bi/multilingual learners for short).

We do this by rejecting deficit-based framings of bi/multilingual students. Deficit framings ignore or treat as problems what kids know and can do, viewing standard school learning objectives as the only or primary learning of value. Instead, we bring teachers and researchers together to explore how schools can build on students' strengths and experiences. We draw on three lenses for design, implementation, and research of CS-integrated curriculum

## Framing our Approach

Our approach to curriculum design, implementation, and research aims to build on and sustain the language practices, identities, and communities of bi/multilingual learners, understanding these populations often intersect with Black, Latinx, indigenous, immigrant, LGBTQIA+, and dis/ability communities -- in line with other culturally sustaining approaches (Paris & Alim, 2017).





We see a world where bi/multilingual learners are encouraged to use their language and code to make meaning, express, critique, and contribute to meaningful conversations, empowering them, and ultimately transforming CS fields and education.

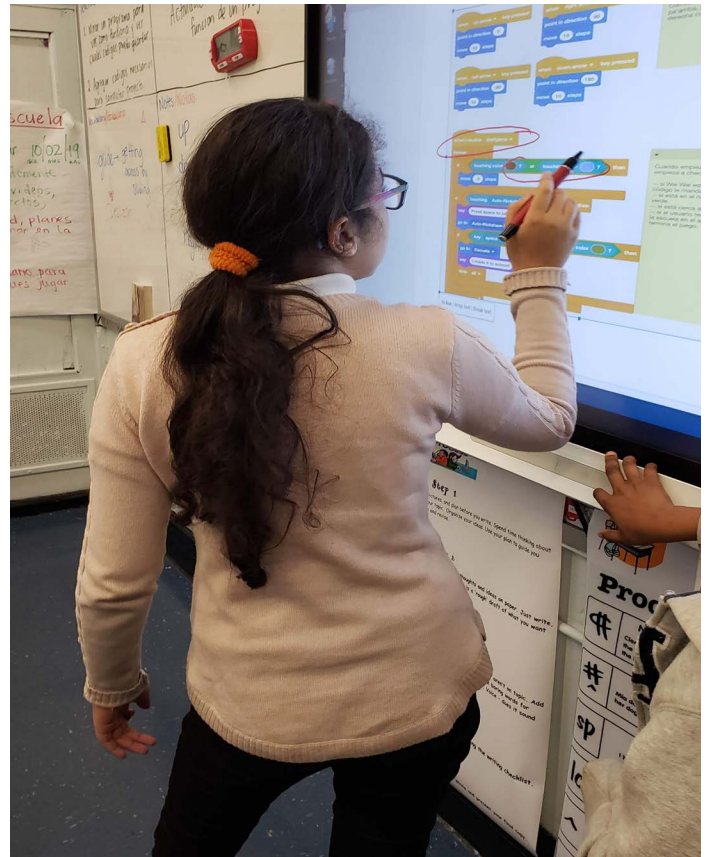
Schooling has traditionally positioned many bi/multilingual learners as strugglers, and language policing multiplies the marginalization of those who are Black, Latinx, indigenous, people of color, immigrants, LGBTQIA+ students, and students with dis/abilities. We instead treat students' varying language backgrounds as a strength.

To promote bi/multilingual learners' meaningful participation in CS Ed, we start with the idea that systems, not students, need to change.

Traditional approaches to computer science are not designed with bi/multilingual learners in mind, and systems are often set up to use differences in language backgrounds to keep them from succeeding.

Our project focuses particularly on the experiences of bi/multilingual learners in CS education, but our work might be more broadly applied. Language mediates how all kids learn computer science. We know that bi/multilingual learners are not the only students who are marginalized for the way they use language (García, 2009; García & Otheguy, 2017) -- for example, students who speak Black varieties of English, students with dyslexia, and students who read with Braille may also be marginalized in different ways for their language use, due to racism and ableism. Building on students' strengths as language-users and meaning-makers might also benefit these students. Even among speakers of "Standard American English," language and discourse help gatekeepers maintain a biased status quo in the CS "clubhouse" (Margolis & Fisher, 2003) through the use of inside jokes and needless jargon (Gee, 2015; Noble & Roberts, 2019). Elevating students' marginalized language practices in CS Ed can open the door to everyone's full talents and participation and helps push back against systems that can keep people out.

“We see a world where bi/multilingual learners are encouraged to use their language and code.”



## Our Guiding Theory

Three theories guide our approach to supporting bi/multilingual students in CS education.

## Translanguaging

Translanguaging is a theory from bilingual education that describes what people do when they use all of their language and communication resources -- including oral language, gestures, drawings, and others -- to make meaning, learn, and express themselves (García & Li Wei, 2014). We are guided by translanguaging pedagogy, which places value on the emerging language and social practices that bi/multilingual learners bring and construct in classrooms (Espinosa & Ascenzi-Moreno, forthcoming). Taking up a translanguaging lens elevates the language practices of traditionally marginalized students, promoting equity in CS education.

Thanks to legacies of racism, classism, and xenophobia, many schools either position emergent bilinguals' multilingual and multimodal resources as deficient or ignore them. Even though society has constructed standard named language categories like "English" and "Spanish," translanguaging theory tells us

that when people communicate they don't compartmentalize their ideas or intelligence in boxes with labels. People draw on a range of semiotic resources including linguistic, social, embodied, and technological methods to communicate, disrupting the idea that only one type of language or modality counts as a legitimate way to think and learn. All people translanguage, but the translanguaging of bi/multilinguals and other marginalized groups is especially stigmatized in society (Otheguy et al., 2015).

This disruptive core of translanguaging theory can be implemented in translanguaging pedagogy (García et al., 2017). This form of pedagogy challenges monolingual instructional practices and policies. It also offers possibilities for learning experiences that build from bi/multilinguals' diverse, dynamic, and expanding repertoires. Translanguaging pedagogy does not jettison goals of learning to communicate in ways that can bring students power--whether that means passing an exam or impressing an interviewer. Instead, it promotes critical thinking about language practices (Seltzer, 2019; Flores & Rosa, 2015) and insists that all of a learner's language background should be embraced as resources for this and all other learning goals.





“Computer scientists should appreciate virtuosic works of programming just like virtuosic works of literature, and . . . treating programs as not only functional but expressive creations would allow real progress in computer science.”

## Literate Programming

Embracing a broad range of literacy practices is not only an idea from bilingual education; this idea also has a long history in computer science itself. Prominent computer scientist Donald Knuth coined the phrase literate programming (Knuth, 1984) to emphasize that computer code (programs) are meant to be read and written by people, and not just computers. Knuth advocated that computer scientists should appreciate virtuosic works of programming just like virtuosic works of literature, and that by treating programs as not only functional but expressive creations would allow real progress in computer science.

Incorporating not just writing of programs, but also reading and engaging with programs, helps create space for novices to participate in computer science. It also helps reduce perceptions of programming as the domain of an elite few, what has been metaphorically called a programming “priesthood” (Backus, 1980; Doctorow, 2009; Maz, 2017; Nelson, 1973; Sabelli, 1998). Engaging critically and reflectively with software in these ways can help improve equity in computer science.

Making programming more literate affects equity in two important ways. First, just like translanguaging pedagogy, it takes an asset based approach in which all of a student’s linguistic resources can be used to leverage into new literacies, i.e., using human language and pseudocode to leverage into executable code. Second, literate programming conceives of programming as part of a dialogue, in which both reading and writing serve as means to participate in a community of discourse. This opens up the possibility that code can be read critically, which is an important part of empowerment. Yasmin Kafai, echoing Paulo Friere’s assertion that “reading the word is reading the world,” argued that computational literacy including what she calls “computational participation” is vital to fully participate in understanding, changing and remaking a world mediated by computation (Kafai, 2016).





## Syncretic Computational Literacies

As we connect our theories of translanguaging pedagogy with the literate programming approach, we always pose the question: “What conversation is code a part of?”

To build on bi/multilingual students’ language practices, identities, and motivations, we aim to foster meaningful conversations in classrooms, drawing on three areas:

### Computational Literacies:

The real-world conversations students can use code and computing to take part in, including not only the way professional programmers might talk to each other, but also other computer science literacies that underpin computing in a variety of settings.

### Disciplinary Literacies:

Ways of reading, writing, creating, and interacting with the world from the subject areas, such as scientific discourse in science class, literary discourse in a language arts class, and so on.

### Community Literacies:

Ways of reading, writing, creating and interacting with the world learned from friends, family, and other communities.

We call this merging together of practices syncretic computational literacies. The term “syncretic” helps us highlight that when people bring practices from different realms together, we create new kinds of literacies out of the tensions and sparks that result, transforming and improving what and how we learn (Gutiérrez, 2014). Designing and implementing syncretic computational conversations in classrooms promotes equity in CS education. Doing so breaks down traditional boundaries between school disciplines and

communities that have systematically marginalized bi/multilingual learners.

School curriculum often undermines and devalues the knowledge of bi/multilingual learners’ communities. As CS education becomes a universal part of the K-12 schooling core, practitioners have the opportunity to challenge this state of affairs. Instead of constructing walls and boundaries around the field, practitioners can view and work towards syncretic goals for CS education, valuing community knowledge as it overlaps with and exists alongside knowledge from computing and the disciplines.

Looking at CS from a syncretic perspective means drawing inspiration for learning environments from the many conversations about, with, and through code that occur in spaces beyond formal CS classes and professional programming jobs. It means noticing and surfacing the tensions between the different ways of knowing, computing and using language that come together in these conversations.

Having syncretic conversations empowers students to use code to serve their communities, push-back against inequitable computing practices, and support their growth and identity development.

“When people bring practices from different realms together, we create new kinds of literacies out of the tensions and sparks that result.”



“We call this merging together of practices syncretic computational literacies.”

**Community**



**Computational**



**Disciplinary**



## Our Activities

PiLa-CS offers up an approach to incorporating language and computer science into subject area curricula to promote the participation of bi/multilingual learners and other traditionally marginalized students in CS for All.

As a research-practice partnership, we study questions relevant to practice, like how multilingual learners use language in learning computer science, and how teachers build on students' communities, interests, and language when teaching computer science. Our practice goals include developing models for supporting learners and educators to incorporate computer science in the classroom.

We do not create canned curriculum, in recognition that translanguaging pedagogy requires teachers to acknowledge and build on the linguistic backgrounds of their students and to adapt curricula to meet those students' needs. Rather, in our project, teachers and researchers work together to co-design curriculum and learning environments that reflect the diverse communities of bi/multilingual learners and promote their participation.

Thus far, we have evidence that partner teachers:

- Provide multiple entry points for students to use language practices from home, community, and school to communicate about, with, and through code.
- Notice, welcome, and expand students' language repertoires, even if the teacher doesn't speak a student's language.
- Incorporate code into conversations they and their communities are already having, deepening those conversations.

Our co-design efforts feed into design of professional development materials and activities that allow educators to take a translanguaging approach in their own contexts.



## Our Research Practice Partnership

With a research-practice partnership (Penuel et al., 2015) approach inspired by design-based implementation research (Fishman et al., 2013), we aim to challenge traditional extractive models of education research by bringing researchers and practitioners together. All are invited to set, implement, and disseminate research and practice agendas. We take time to consider how dynamics of positionality -- race, gender, affiliations, language -- shape our work together over time. We aim to build reciprocal, mutually beneficial relationships that can be responsive to school and classroom context (Bang & Vossoughi, 2016).

Our values around equity surface in how our RPP works together:

- Participation looks different for all members, but all participation is valued.
- Teachers are co-designers of curriculum, not just "implementers" of a canned approach.
- Researchers' practices are as much under study as teachers' and students'.
- Teachers present curriculum and findings to both academic and professional audiences.





## Suggested Citation

Vogel, S., Hoadley, C., Ascenzi-Moreno, L., Ma, J., James, S., & Palacios, O. (2020, Oct). Our Approach. Participating in Literacies and Computer Science (PiLaCS). <https://nyuscholars.nyu.edu/en/publications/our-approach>

## References

- Backus, J. (1980). Programming in America in the 1950s—Some personal impressions. In N. Metropolis, J. Howlett, & G.-C. Rota (Eds.), *A history of computing in the twentieth century: A collection of essays* (pp. 125-136). Academic Press.
- Bang, M., & Vossoughi, S. (2016). Participatory Design Research and Educational Justice: Studying Learning and Relations Within Social Change Making. *Cognition and Instruction*, 34(3), 173–193. <https://doi.org/10.1080/07370008.2016.1181879>
- Doctorow, C. (2009, 12 March). The high priests of IT—And the heretics. *Harvard Business Review*. <https://hbr.org/2009/03/the-high-priests-of-it>
- Espinosa, C., & Ascenzi-Moreno, L. (Forthcoming). *Rooted in Strength: Growing Multilingual Readers and Writers, K-5*.
- Fishman, B. J., Penuel, W. R., Allen, A.-R., Cheng, B. H., & Sabelli, N. (2013). Design-based implementation research: An emerging model for transforming the relationship of research and practice. *National Society for the Study of Education*, 112(2), 136–156.
- Flores, N., & Rosa, J. (2015). Undoing Appropriateness: Raciolinguistic Ideologies and Language Diversity in Education. *Harvard Educational Review*, 85(2), 149–171. <https://doi.org/10.17763/0017-8055.85.2.149>
- García, O. (2009). *Bilingual Education in the 21st Century: A Global Perspective*. Wiley/Blackwell.
- García, O., Johnson, S., & Seltzer, K. (2017). *The Translanguaging Classroom: Leveraging Student Bilingualism for Learning*. Caslon Publishing.
- García, O., & Li Wei. (2014). *Translanguaging: Language, bilingualism and education*. Palgrave Macmillan Pivot.
- García, O., & Otheguy, R. (2017). Interrogating the Language Gap of Young Bilingual and Bidialectal Students. *International Multilingual Research Journal*, 11(1), 52–65. <https://doi.org/10.1080/19313152.2016.1258190>
- Gee, J. (2015). *Social Linguistics and Literacies: Ideology in Discourses (Fifth Edition)*. Routledge. <https://doi-org.proxy.library.nyu.edu/10.4324/9781315722511>
- Gutiérrez, K. D. (2014). Integrative research review: Syncretic approaches to literacy learning: Leveraging horizontal knowledge and expertise. In P. J. Dunston, L. B. Gambrell, K. Headley, S. K. Fullerton, & P. M. Stecker (Eds.), *63rd Literacy Research Association Yearbook* (pp. 48–60). [http://www.academia.edu/28876369/Syncretic\\_Approaches\\_to\\_Literacy\\_Learning\\_-\\_Leveraging\\_Horizontal\\_Knowledge\\_and\\_Expertise.pdf](http://www.academia.edu/28876369/Syncretic_Approaches_to_Literacy_Learning_-_Leveraging_Horizontal_Knowledge_and_Expertise.pdf)
- Kafai, Y. B. (2016). From computational thinking to computational participation in K--12 education. *Communications of the ACM*, 59(8), 26–27. <https://doi.org/10.1145/2955114>







## References (cont'd)

- Knuth, D. E. (1984). Literate Programming. *The Computer Journal*, 27(2), 97–111. <https://doi.org/10.1093/comjnl/27.2.97>
- Lesser, M. (Aug 5). Dwelling in the Borderlands. <https://www.stitcher.com/s?eid=76741643>
- Margolis, J., & Fisher, A. (2003). *Unlocking the Clubhouse: Women in Computing*. MIT Press.
- Maz, A. (2017, 5 December). A priesthood of programmers. *Jacobite*, (December 2017). <https://jacobitemag.com/2017/12/05/a-priesthood-of-programmers/>
- Nelson, T. H. (1973, June 4-8). A conceptual framework for man-machine everything Proceedings of the National Computer Conference and Exposition - AFIPS '73, New York. <http://doi.org/10.1145/1499586.1499776>
- Otheguy, R., García, O., & Reid, W. (2015). Clarifying translanguaging and deconstructing named languages: A perspective from linguistics. *Applied Linguistics Review*, 6(3), 281–307. <https://doi.org/10.1515/applirev-2015-0014>
- Penuel, W. R., Allen, A.-R., Coburn, C. E., & Farrell, C. (2015). Conceptualizing research–practice partnerships as joint work at boundaries. *Journal of Education for Students Placed at Risk (JESPAR)*, 20(1–2), 182–197.
- Sabelli, N. H. (1998). We are no longer a priesthood. *Communications of the ACM*, 41(1), 20–21. <http://doi.org/10.1145/268092.268100>

