Exploring Bilingual Coding for Inclusive Computer Science Learning

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Anonymous Author(s)
anonymous@example.com
Anonymous Institution
Anonymous Country

Abstract

As computing education expands globally, the dominance of English in programming languages can present barriers for students who are not native English speakers. This research work investigates whether language matters in programming by examining how a bilingual programming workshop influences student attitudes, perceived understanding and comfort. Sixty bilingual participants from a University in USA, 40 with little to no prior programming experience and 20 experienced programmers completed a two-hour workshop using the multilingual Hedy platform. Pre- and post-surveys, including the validated Attitudes Toward Computing Scale, measured changes in confidence, enjoyment, motivation, identity, and perceived usefulness. Additionally, participants were given open-ended questions on their expreinces. All students showed improvements across most constructs, but inexperienced students gained more in confidence and enjoyment, while experienced students showed smaller gains and preferred coding in English. Qualitative responses revealed that while most students felt English supported clarity, bilingual students more often reported that using both languages helped with conceptual understanding, despite some translation challenges. Overall, the findings suggest that bilingual coding environments may enhance engagement and accessibility for beginners, though language preferences and prior experience continue to shape how students interact with code.

SIGCSE 2026, Portland, OR, USA

ACM ISBN 978-1-4503-XXXX-X/26/03
https://doi.org/10.1145/XXXXXXX.XXXXXXX

CCS Concepts

• **Social and professional topics → Computer science education**.

Keywords

Bilingualism, Culture, Language, Programming, Broadening Participation, K-12

ACM Reference Format:
Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), Anonymous Author(s), and Anonymous Author(s). 2026. Exploring Bilingual Coding for Inclusive Computer Science Learning. In *57th ACM Technical Symposium on Computer Science Education (SIGCSE 2026), March 11–14, 2026, Portland, OR, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/XXXXXXX.XXXXXXX

1 Motivation and Background

Language plays a critical yet often overlooked role in computer science education. Students who are not native English speakers often face linguistic barriers that affect their comprehension and confidence in programming [1–3]. This study investigates how bilingual coding, using both English and a native language, affects students' engagement, attitudes, and perceived understanding of programming concepts. By leveraging Hedy, a multilingual programming environment, we aimed to explore whether allowing students to code bilingually can enhance inclusivity and conceptual learning, especially among beginners.

Specifically, we compare two groups of inexperienced students, those who coded bilingually versus those who coded in English. We then compare inexperienced students to experienced students across the same four dimensions. Our research questions are:

- **RQ1**: How do students' attitudes toward programming change after coding bilingually?
- **RQ2**: What language-related challenges do students report?
- **RQ3**: How does language affect students' perceived understanding of programming?

- **RQ4**: How comfortable do students feel coding in different languages?

2 Methods

Sixty bilingual university students participated in a two-hour programming workshop that involved coding tasks using Hedy, a gradual and multilingual programming environment. Participants were evenly split between experienced (n=20) and inexperienced (n=40) programmers, with experience defined as 6+ months of coding. The study employed a pre–post survey design using the Attitudes Toward Computing Scale Wanzer et al. [4] to measure students' enjoyment, confidence, and perceived relevance of programming. Quantitative data were analyzed using paired-sample t-tests to assess changes between pre- and post-survey responses, and independent-sample t-tests to compare subgroup differences between beginners and experienced programmers. Qualitative responses from open-ended questions were analyzed thematically to identify language-related benefits and challenges. (Future work will employ a Linear Mixed Effects model to better account for within-participant variation.)

During the workshop, students followed a use–modify–create structure through Hedy's levels (1–5), covering input/output, variables, and conditionals. Research assistants provided support, and the platform offered real-time feedback. As shown in Figure 1, students interacted with a bilingual programming environment during the workshop creating a restaurant system. Participants worked at their own pace, and those who completed all tasks received an Amazon voucher.

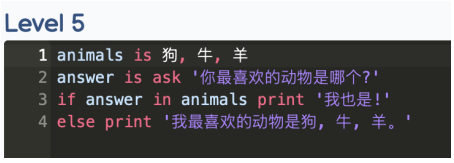

Figure 1: Sample Bilingual programme

2.1 Key findings

We present the results of our study, highlighting both quantitative measures of student attitudes and identities as well as qualitative insights from open-ended responses, organized by research question and participant characteristics.

Quantitative Highlights:

Across the 60 participants, the programming workshop led to significant gains in confidence, computing identity, enjoyment, and motivation. Within inexperienced students, paired-sample t-tests and Wilcoxon signed-rank tests revealed significant pre-to-post gains in programming confidence , computing identity, enjoyment, and motivation, while perceived usefulness did not change significantly. Inexperienced participants exhibited significantly greater gains in programming confidence than experienced participants, with enjoyment approaching significance. Differences in computing identity, motivation, and perceived usefulness were not statistically significant, though effect sizes suggest modest trends favoring inexperienced learners. Overall, the intervention (regardless of language) appeared especially impactful for novices, boosting confidence and engagement, while experienced coders showed smaller gains, potentially due to ceiling effects or prior familiarity. When comparing bilingual vs. English-only students, there were no statistically significant differences in any constructs (confidence, identity, enjoyment, motivation, or perceived usefulness). Effect sizes were small.

Qualitative Highlights: Many students who coded bilingual noted that coding in both languages helped them understand programming concepts, though challenges such as translation inconsistencies and limited bilingual documentation were common. Experienced students were more likely to report difficulties related to transfer from prior coding habits and confusion with native coding terms, whereas inexperienced students struggled more with syntax, conceptual understanding, and translation clarity. When comparing bilingual versus English-only inexperienced students, bilingual participants reported greater conceptual benefits and language-mediated flexibility, despite encountering more translation and platform-related challenges. English-only students generally emphasized clarity and ease of understanding in English, though fewer reported conceptual gains from language support. Across experience levels, coding in English was more comfortable for experienced students, while bilingual inexperienced students demonstrated greater openness to using both languages.

3 Discussion

The findings suggest that bilingual coding can enhance engagement and confidence, particularly among beginner programmers. While quantitative outcomes were similar across groups, qualitative insights highlight the potential of multilingual environments to reduce intimidation and increase accessibility.

The study reinforces the importance of culturally and linguistically responsive approaches in computer science education. Future work will expand the study to longer interventions and apply a Linear Mixed Effects model for more robust within-subject analysis.

4 Conclusions and Implications

Bilingual coding environments may support inclusion by lowering language barriers and enabling learners to connect programming concepts with familiar linguistic contexts. As computer science education expands globally, embracing linguistic diversity can open new pathways for equitable participation and learning.

References

[1] Pedro Guillermo Feijóo-García, Keith McNamara, and Jacob Stuart. 2020. The effects of native language on block-based programming introduction: A work in progress with hispanic population. In *2020 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, Vol. 1. IEEE, 1–2.

[2] Adalbert Gerald Soosai Raj, Kasama Ketsuriyonk, Jignesh M Patel, and Richard Halverson. 2018. Does native language play a role in learning a programming language?. In *Proceedings of the 49th ACM technical symposium on computer science education*. 417–422.

[3] Sara Vogel, Christopher Hoadley, Laura Ascenzi-Moreno, and Kate Menken. 2019. The role of translanguaging in computational literacies: Documenting middle school bilinguals' practices in computer science integrated units. In *Proceedings of the 50th ACM technical symposium on computer science education*. 1164–1170.

[4] Dana Wanzer, Tom McKlin, Doug Edwards, Jason Freeman, and Brian Magerko. 2019. Assessing the attitudes towards computing scale: A survey validation study. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. 859–865.