Knowledge Component-Driven Alignment of CS1 Textbooks and Exercises

Samantha Smith
University of California, Berkeley
Berkeley, United States
happysammie6@berkeley.edu

Arun-Balajiee Lekshmi Narayanan
University of Pittsburgh
Pittsburgh, United States
arl122@pitt.edu

Anurata Prabha Hridi
North Carolina State University
Raleigh, United States
aphridi@ncsu.edu

Rafaella Sampaio de Alencar
University of Pittsburgh
Pittsburgh, United States
ras555@pitt.edu

Bita Akram
North Carolina State University
Raleigh, United States
bakram@ncsu.edu

Arto Hellas
Aalto University
Espoo, Finland
arto.hellas@aalto.fi

Juho Leinonen
Aalto University
Espoo, Finland
juho.2.leinonen@aalto.fi

Peter Brusilovsky
University of Pittsburgh
Pittsburgh, United States
peterb@pitt.edu

Narges Norouzi
University of California, Berkeley
Berkeley, United States
norouzi@berkeley.edu

Abstract

We present a reproducible pipeline that aligns CS1 textbook sections with problems from a public dataset via a Knowledge Component (KC) -a single conceptual skill required for problem solving- ontology. It assigns KCs to sections and problems, respects the prerequisite order to avoid inserting problems too early, and generates tips for not-yet-taught concepts. We evaluate three KC assignment strategies: embedding-only, embedding with a Large Language Model (LLM) tie-breaker, and direct LLM assignment. We find direct assignment matches or exceeds human annotators. Our results show that constrained LLMs can enrich CS1 textbooks with curriculum-aware practice problems.

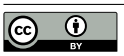
CCS Concepts

• **Social and professional topics → Computing education**.

Keywords

Knowledge Components, Large Language Model, CS1, Textbook Augmentation, Curriculum Alignment

ACM Reference Format:
Samantha Smith, Arun-Balajiee Lekshmi Narayanan, Anurata Prabha Hridi, Rafaella Sampaio de Alencar, Bita Akram, Arto Hellas, Juho Leinonen, Peter Brusilovsky, and Narges Norouzi. 2026. Knowledge Component-Driven Alignment of CS1 Textbooks and Exercises. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.2 (SIGCSE TS 2026), February 18–21, 2026, St. Louis, MO, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3770761.3777243



SIGCSE TS 2026, St. Louis, MO, USA

ACM ISBN 979-8-4007-2255-4/2026/02
https://doi.org/10.1145/3770761.3777243

1 Introduction and Related Work

LLMs enable new opportunities to align practice problems with curricula at scale. Recent work shows they can extract core concepts and prerequisites from course materials [2], motivating KC-based approaches. Earlier example insertion methods relied on shallow features and ignored prerequisite order [4]. Meanwhile, CS education research highlights growing use of LLMs for tutoring and classification, with a need for interpretable controls [5, 7].

This work contributes a reproducible pipeline that assigns KCs grounded in established CS1 literature [1, 3] to both textbook sections and CS1 problems from the public dataset, Standards, Protocols, and Learning Infrastructure for Computing Education (SPLICE) [6]. The system selects problems without violating the textbook's prerequisite order and generates short tips when problems involve not-yet-taught concepts that are not central to the solution. By controlling the order of problems, the pipeline demonstrates reliable, curriculum-aware alignment of practice problems with textbooks. In this poster, we examine two method-oriented questions:

RQ1: How can a unified KC ontology enable automatic, prerequisite-aware matching between CS1 textbook sections and external problem banks?

RQ2: Among practical labeling strategies, which yields the most consistent KC assignments relative to multiple human annotators?

The work presented in this paper was partially supported by the National Science Foundation under Grant No. 2426839, 2426838, 2426837.

2 Method

(1) **KC Ontology and Prerequisite Ordering:** We construct a CS1 KC list from established literature [1, 3] and order it according to the sequence in the textbook.
(2) **Human Baseline:** Three annotators labeled SPLICE problems and textbook sections with KCs. Agreement was measured using the standard, triple, and quadruple Dice index D, D_3, D_4.

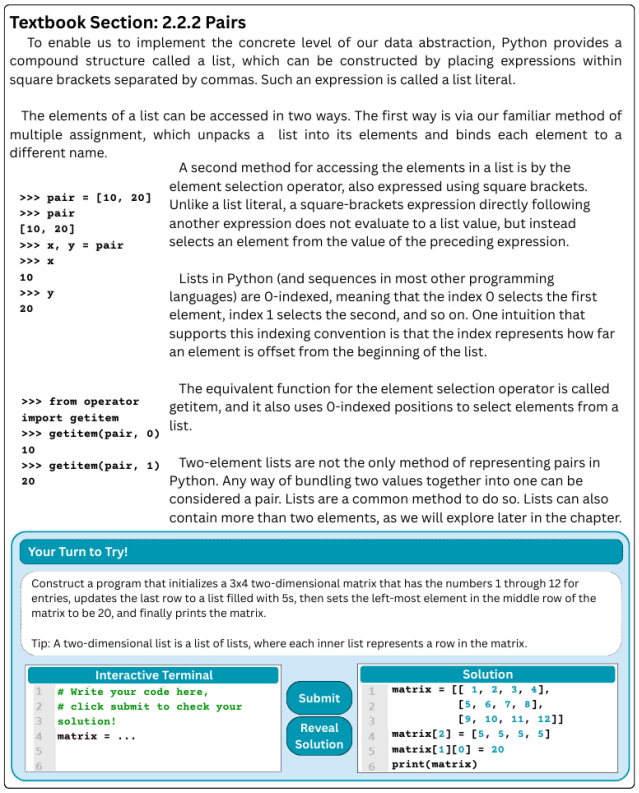


Figure 1: Insertion of an example (blue) in textbook (white). The activity instruction includes tip(s) for not-yet-taught KCs (e.g., two-dimensional list).

(3) **Textbook KC Extraction:** We tested three assignment strategies with increasing LLM involvement. We used MiniLM-L6-v2 for embeddings, GPT-4o for KC assignment.
   (a) *Algorithm 1 – Embedding-Only:* Assign KCs by top cosine similarity match of textbook section and KCs.
   (b) *Algorithm 2 – Embedding + LLM Tie-Breaker:* Get the top three embedding matches, ask the LLM to choose the best.
   (c) *Algorithm 3 – Direct LLM Assignment:* Ask the LLM to assign KCs directly.
(4) **Problem KC Extraction:** Each SPLICE problem is processed by the LLM to identify relevant KCs and a single "target KC".
(5) **Matching:** Candidate problems are filtered by prerequisite order; an LLM judge then selects the best match and, when needed, generates a short tip to bridge gaps.

3 Results and Discussion

RQ1 (KCs and prerequisite-aware matching). Human annotators had moderate agreement on SPLICE problems ($D \in [0.35, 0.44]$; $D_3 = 0.40$). Including LLM labels in the Dice calculation boosted inter-labeler agreement to $D_3 \in [0.50, 0.56]$, suggesting automated assignment can act as a stabilizing "fourth annotator." For textbook sections, human agreement was higher ($D \in [0.79, 0.81]$; $D_3 = 0.72$), and SPLICE problems were successfully inserted without breaking prerequisite order, generating short tips when needed (Figure 1).

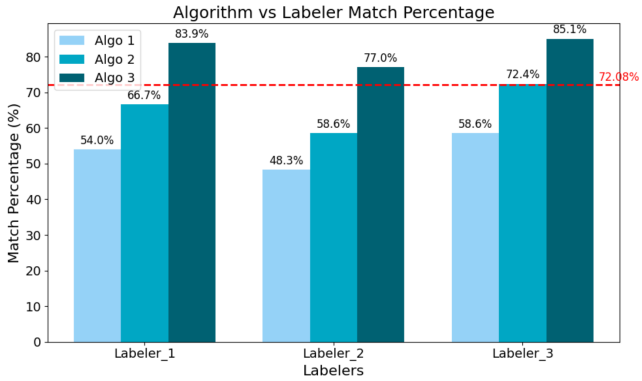


Figure 2: KC extraction vs human baseline (red = inter-labeler Dice)

RQ2 (labeling strategy consistency). Figure 2 shows that algorithmic match quality increases with LLM involvement: embedding-only matched ~60% of human labels, embedding + LLM tie-breaker ~75%, and direct LLM assignment ~88%. These rates meet or exceed individual human consistency (~72%), demonstrating that constrained LLM assignment produces reliable KC labels; moreover, D_4 analysis confirmed significant four-way agreement in 81.6% of sections ($p < 0.05$, $D_4 = 0.65$).

Together, these findings show that the KC-first pipeline respects prerequisite order (MQ1) and that direct LLM assignment is the most dependable strategy for KC extraction (MQ2).

4 Limitations and Future Work

This work centers on pipeline design, not classroom use. Manual prerequisite ordering may not transfer across curricula, and we assume one "target KC" per problem. Future work will extend to other CS1 variants, textbooks, and problem types (e.g. natural language and conceptual exercises), and will evaluate at scale with students and instructors to assess pedagogical value and generalizability.

References

[1] Kathryn Cunningham, Sarah Heckman, Nathan Sprague, et al. 2024. A Computing Core Concepts Inventory: Developing and Validating a Knowledge Instrument for CS1. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education (SIGCSE)*. doi:10.1145/3664191
[2] Ngoc Luyen Le and Marie-Hélène Abel. 2025. How Well Do LLMs Predict Prerequisite Skills? Zero-Shot Comparison to Expert-Defined Concepts. (2025). arXiv:2507.18479 [cs.IR] https://arxiv.org/abs/2507.18479
[3] Raymond Lister and Donna Teague. 2010. Further Evidence of a Relationship Between Spatial Ability and Success in Introductory Programming Courses. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education (SIGCSE)*. 186–190. doi:10.1145/1734263.1734297
[4] Jeffrey Matayoshi and Christopher Lechuga. 2020. Automated Matching of ITS Problems with Textbook Content. In *Proceedings of the iTextbooks Workshop at AIED*. 17–28.
[5] Nishat Raihan, Mohammed Latif Siddiq, Joanna C. S. Santos, and Marcos Zampieri. 2025. Large Language Models in Computer Science Education: A Systematic Literature Review. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*. 938–944.
[6] Cliff Shaffer, Peter Brusilovsky, Ken Koedinger, Thomas Price, Tiffany Barnes, and Behrooz Mostafavi. 2024. Ninth SPLICE Workshop on Technology and Data Infrastructure for CS Education Research. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 1904. doi:10.1145/3626253.3633431
[7] Thomas Y. Yeh, Karena Tran, Ge Gao, Tyler Yu, Wai On Fong, and Tzu-Yi Chen. 2025. Bridging Novice Programmers and LLMs with Interactivity. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*. 1295–1301.