













## REFERENCES

- [1] Vincent Alevan, Ido Roll, Bruce M. McLaren, and Kenneth R. Koedinger. 2016. Help Helps, But Only So Much: Research on Help Seeking with Intelligent Tutoring Systems. *International Journal of Artificial Intelligence in Education* 26, 1 (2016), 1–19.
- [2] John R. Anderson, Frederick G. Conrad, and Albert T. Corbett. 1989. Skill Acquisition and the LISP tutor. *Cognitive Science* 13, 4 (1989), 467–505.
- [3] Michael Ball. 2018. *Lambda: An Autograder for Snap!*. Technical Report. Electrical Engineering and Computer Sciences University of California at Berkeley. <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2018/EECS-2018-2.pdf>
- [4] Joseph E Beck, Kai-min Chang, Jack Mostow, and Albert Corbett. 2008. Does help help? Introducing the Bayesian Evaluation and Assessment methodology. In *International Conference on Intelligent Tutoring Systems*. Springer, 383–394.
- [5] Tara S. Behrend, David J. Sharek, Adam W. Meade, and Eric N. Wiebe. 2011. The viability of crowdsourcing for survey research. *Behavior Research Methods* 43, 3 (25 Mar 2011), 800.
- [6] Jens Bennedsen and Michael E. Caspersen. 2007. Failure rates in introductory programming. *ACM SIGCSE Bulletin* 39, 2 (2007), 32. <https://doi.org/10.1145/1272848.1272879>
- [7] Xianglei Chen and Matthew Soldner. 2013. *STEM Attrition: College Students' Paths Into and Out of STEM Fields*. Technical Report. National Center for Education Statistics, Institute of Education Sciences, U.S. Department of Education. <http://nces.ed.gov/pubs2014/2014001rev.pdf>
- [8] Albert T. Corbett. 2001. Cognitive Computer Tutors: Solving the Two-Sigma Problem. In *Proceedings of the International Conference on User Modeling*. Springer, 137–147.
- [9] Albert T Corbett and John R Anderson. 2001. Locus of feedback control in computer-based tutoring: Impact on learning rate, achievement and attitudes. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 245–252.
- [10] Davide Fossati, Barbara Di Eugenio, Stellan Ohlsson, Christopher Brown, and Lin Chen. 2010. Generating proactive feedback to help students stay on track. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 6095 LNCS, PART 2 (2010), 315–317.
- [11] Davide Fossati, Barbara Di Eugenio, Stellan Ohlsson, Christopher Brown, and Lin Chen. 2015. Data Driven Automatic Feedback Generation in the iList Intelligent Tutoring System. *Technology, Instruction, Cognition and Learning* 10, 1 (2015), 5–26.
- [12] Dan Garcia, Brian Harvey, and Tiffany Barnes. 2015. The Beauty and Joy of Computing. *ACM Inroads* 6, 4 (2015), 71–79.
- [13] Alex Gerdes, Bastiaan Heeren, Johan Jeuring, and L. Thomas van Binsbergen. 2017. Ask-Elle: an Adaptable Programming Tutor for Haskell Giving Automated Feedback. *International Journal of Artificial Intelligence in Education* 27, 1 (2017), 1–36.
- [14] Brian Harvey, Daniel Garcia, Josh Paley, and Luke Segars. 2012. Snap!(build your own blocks). In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*. ACM, 662–662.
- [15] Andrew Head, Elena Glassman, Gustavo Soares, Ryo Suzuki, Lucas Figueredo, Loris D'Antoni, and Björn Hartmann. 2017. Writing Reusable Code Feedback at Scale with Mixed-Initiative Program Synthesis. In *Proceedings of the ACM Conference on Learning @ Scale*. ACM, 89–98.
- [16] Jay Holland, Antonija Mitrovic, and Brent Martin. 2009. J-LATTE: a Constraint-based Tutor for Java. In *Proceedings of the International Conference on Computers in Education*. University of Canterbury. Computer Science and Software Engineering, 142–146.
- [17] Shalini Kaleeswaran, Anirudh Santhiar, Aditya Kanade, and Sumit Gulwani. 2016. Semi-Supervised Verified Feedback Generation. *CoRR* (2016), 739–750.
- [18] Hieke Keuning, Johan Jeuring, and Bastiaan Heeren. 2016. Towards a Systematic Review of Automated Feedback Generation for Programming Exercises. In *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '16*. ACM, 41–46. <https://doi.org/10.1145/2899415.2899422>
- [19] Aniket Kittur, Ed H Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with Mechanical Turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, 453–456.
- [20] Nguyen-Thinh Le, Wolfgang Menzel, and Niels Pinkwart. 2009. Evaluation of a constraint-based homework assistance system for logic programming. *Proceedings of the 17th International Conference on Computers in Education* (2009), 51–58.
- [21] Victor J. Marin, Tobin Pereira, Srinivas Sridharan, and Carlos R. Rivero. 2017. Automated personalized feedback in introductory Java programming MOOCs. *Proceedings - International Conference on Data Engineering* August (2017), 1259–1270.
- [22] Danielle S. McNamara. 2017. Self-Explanation and Reading Strategy Training (SERT) Improves Low-Knowledge Students' Science Course Performance. *Discourse Processes* (2017).
- [23] Antonija Mitrovic. 1998. A knowledge-based teaching system for SQL. In *Proceedings of ED-MEDIA*, Vol. 98. 1027–1032.
- [24] Antonija Mitrovic, Brent Martin, and Pramuditha Suraweera. 2007. Intelligent tutors for all: Constraint-based modeling methodology, systems and authoring. *IEEE Intelligent Systems* 22 (2007), 38–45.
- [25] Benjamin Paaßen, Barbara Hammer, Thomas William Price, Tiffany Barnes, Sebastian Gross, and Niels Pinkwart. 2017. The Continuous Hint Factory - Providing Hints in Vast and Sparsely Populated Edit Distance Spaces. *arXiv preprint arXiv:1708.06564* 10, 1 (2017), 1–35. [arXiv:1708.06564](https://arxiv.org/abs/1708.06564)
- [26] Daniel Perelman, Sumit Gulwani, and Dan Grossman. 2014. Test-driven synthesis for automated feedback for introductory computer science assignments. *Proceedings of Data Mining for Educational Assessment and Feedback (ASSESS 2014)* (2014).
- [27] C Piech, J Huang, A Nguyen, M Phulsuksombati, M Sahami, and L Guibas. 2015. Learning program embeddings to propagate feedback on student code. *arXiv preprint arXiv:1505.05969* (2015), 1093–1102. [arXiv:1505.05969v1](https://arxiv.org/abs/1505.05969)
- [28] Thomas W Price, Yihuan Dong, and Dragan Lipovac. 2017. iSnap: towards intelligent tutoring in novice programming environments. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. ACM, 483–488.
- [29] Thomas W Price, Zhongxiu Liu, Veronica Cateté, and Tiffany Barnes. 2017. Factors Influencing Students' Help-Seeking Behavior while Programming with Human and Computer Tutors. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*. ACM, 127–135.
- [30] Thomas W. Price, Joseph J. Williams, and Samiha Marwan. 2019. A Comparison of Two Designs for Automated Programming Hints. *2nd Educational Data Mining in Computer Science Education (CSEDM) Workshop at the International Conference on Learning Analytics and Knowledge (LAK)* (2019).
- [31] Thomas W. Price, Rui Zhi, and Tiffany Barnes. 2017. Evaluation of a Data-driven Feedback Algorithm for Open-ended Programming. In *Proceedings of the International Conference on Educational Data Mining*.
- [32] Thomas W Price, Rui Zhi, and Tiffany Barnes. 2017. Hint generation under uncertainty: The effect of hint quality on help-seeking behavior. In *International Conference on Artificial Intelligence in Education*. Springer, 311–322.
- [33] Kelly Rivers. 2017. *Automated Data-Driven Hint Generation for Learning Programming*. PhD. Carnegie Mellon University.
- [34] Kelly Rivers and Kenneth R. Koedinger. 2017. Data-Driven Hint Generation in Vast Solution Spaces: a Self-Improving Python Programming Tutor. *International Journal of Artificial Intelligence in Education* 27, 1 (2017), 37–64.
- [35] André L. Santos. 2012. An open-ended environment for teaching Java in context. In *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education - ITiCSE '12*. ACM Press, New York, New York, USA, 87.
- [36] Rishabh Singh, Sumit Gulwani, and Armando Solar-Lezama. 2013. Automated feedback generation for introductory programming assignments. *Acm Sigplan Notices* 48, 6 (2013), 15–26.
- [37] Arto Vihavainen, Jonne Airaksinen, and Christopher Watson. 2014. A systematic review of approaches for teaching introductory programming and their influence on success. *Proceedings of the tenth annual conference on International computing education research - ICER '14* (2014), 19–26.
- [38] Christopher Watson and Frederick W B Li. 2014. Failure rates in introductory programming revisited. In *Proceedings of the ACM Conference on Innovation and Technology in Computer Science Education*. ACM, ACM, 39–44.
- [39] Jooyong Yi, Umair Z. Ahmed, Amey Karkare, Shin Hwei Tan, and Abhik Roychoudhury. 2017. A Feasibility Study of Using Automated Program Repair for Introductory Programming Assignments. In *Proceedings of the Joint Meeting on Foundations of Software Engineering*. ACM, 740–751. <https://doi.org/10.1145/3106237.3106262>