Universiteit Leiden
The Netherlands

Variables and variable naming in introductory programming education

Werf, V. van der

Citation
Werf, V. van der. (2025, September 2). *Variables and variable naming in introductory programming education*. Retrieved from https://hdl.handle.net/1887/4259393

Version:	Publisher's Version
License:	Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden
Downloaded from:	https://hdl.handle.net/1887/4259393

Note: To cite this publication please use the final published version (if applicable).

Bibliography

[Abrami et al., 2015] Abrami, P. C., Bernard, R. M., Borokhovski, E., Waddington, D. I., Wade, C. A., and Persson, T. (2015). Strategies for teaching students to think critically: A meta-analysis. *Review of Educational Research*, 85(2):275–314.

[Allamanis et al., 2014] Allamanis, M., Barr, E. T., Bird, C., and Sutton, C. (2014). Learning natural coding conventions. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2014, page 281–293, New York, NY, USA. Association for Computing Machinery.

[Antoniol et al., 2002] Antoniol, G., Canfora, G., Casazza, G., De Lucia, A., and Merlo, E. (2002). Recovering traceability links between code and documentation. *IEEE Transactions on Software Engineering*, 28(10):970–983.

[Arnaoudova et al., 2016] Arnaoudova, V., Di Penta, M., and Antoniol, G. (2016). Linguistic antipatterns: what they are and how developers perceive them. *Empirical Software Engineering*, 21(1):104–158.

[Avidan and Feitelson, 2017] Avidan, E. and Feitelson, D. G. (2017). Effects of variable names on comprehension: An empirical study. In *2017 IEEE/ACM 25th International Conference on Program Comprehension (ICPC)*, pages 55–65.

[Beniamini et al., 2017] Beniamini, G., Gingichashvili, S., Orbach, A. K., and Feitelson, D. G. (2017). Meaningful identifier names: The case of single-letter variables. In *2017 IEEE/ACM 25th International Conference on Program Comprehension (ICPC)*, pages 45–54.

[Bettin and Ott, 2023] Bettin, B. and Ott, L. (2023). Pedagogical prisms: Toward domain isomorphic analogy design for relevance and engagement in computing education. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2023, page 410–416, New York, NY, USA. Association for Computing Machinery.

[Bettin et al., 2023] Bettin, B., Ott, L., and Hiebel, J. (2023). More (sema|meta)phors: Additional perspectives on analogy use from concurrent programming students. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2023, page 166–172, New York, NY, USA. Association for Computing Machinery.

[Biggs and Collins, 1982] Biggs, J. B. and Collins, K. F. (1982). *Evaluating the Quality of Learning. The SOLO Taxonomy.* Elsevier Inc. Publication Title: Educational Psychology.

[Binkley et al., 2009] Binkley, D., Lawrie, D., Maex, S., and Morrell, C. (2009). Identifier length and limited programmer memory. *Science of Computer Programming*, 74(7):430–445.

[Blinman and Cockburn, 2005] Blinman, S. and Cockburn, A. (2005). Program comprehension: Investigating the effects of naming style and documentation. In *AUIC*.

[Börstler et al., 2017] Börstler, J., Störrle, H., Toll, D., van Assema, J., Duran, R., Hooshangi, S., Jeuring, J., Keuning, H., Kleiner, C., and MacKellar, B. (2017). "i know it when i see it": Perceptions of code quality. In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '17, page 389, New York, NY, USA. Association for Computing Machinery.

[Boulay, 1986] Boulay, B. D. (1986). Some difficulties of learning to program. *Journal of Educational Computing Research*, 2(1):57–73.

[Brennan and Resnick, 2012] Brennan, K. and Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *American Educational Research Association*. ISSN: 1860949X.

[Briggs, 2023] Briggs, J. R. (2023). *Python for Kids: A plyaful introduction to programming (2nd edition)*. No Starch Press.

[Brooks, 1983] Brooks, R. (1983). Towards a theory of the comprehension of computer programs. *International Journal of Man-Machine Studies*, 18(6):543–554.

[Busjahn and Schulte, 2013] Busjahn, T. and Schulte, C. (2013). The use of code reading in teaching programming. In *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, Koli Calling '13, page 3–11, New York, NY, USA. Association for Computing Machinery.

[Butler, 2009] Butler, S. (2009). The effect of identifier naming on source code readability and quality. In *Proceedings of the Doctoral Symposium for ESEC/FSE on Doctoral Symposium*, ESEC/FSE Doctoral Symposium '09, page 33–34, New York, NY, USA. Association for Computing Machinery.

[Butler et al., 2015] Butler, S., Wermelinger, M., and Yu, Y. (2015). Investigating naming convention adherence in java references. In *2015 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 41–50.

[Butler et al., 2009] Butler, S., Wermelinger, M., Yu, Y., and Sharp, H. (2009). Relating identifier naming flaws and code quality: An empirical study. In *2009 16th Working Conference on Reverse Engineering*, pages 31–35.

[Butler et al., 2010] Butler, S., Wermelinger, M., Yu, Y., and Sharp, H. (2010). Exploring the influence of identifier names on code quality: An empirical study. In *2010 14th European Conference on Software Maintenance and Reengineering*, pages 156–165.

[Caprile and Tonella, 1999] Caprile, B. and Tonella, P. (1999). Nomen est omen: analyzing the language of function identifiers. In *Sixth Working Conference on Reverse Engineering (Cat. No.PR00303)*, pages 112–122.

[Caprile and Tonella, 2000] Caprile, B. and Tonella, P. (2000). Restructuring program identifier names. In *Proceedings 2000 International Conference on Software Maintenance*, pages 97–107, San Jose, CA, USA. IEEE.

[Cates et al., 2021] Cates, R., Yunik, N., and Feitelson, D. G. (2021). Does code structure affect comprehension? on using and naming intermediate variables. In *2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC)*, pages 118–126.

[Charmaz, 2014] Charmaz, K. (2014). *Constructing grounded theory*. Introducing qualitative methods 181070847. SAGE, London, 2nd edition. edition.

[Chen et al., 2020] Chen, B., Azad, S., Haldar, R., West, M., and Zilles, C. (2020). A validated scoring rubric for explain-in-plain-english questions. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, SIGCSE '20, page 563–569, New York, NY, USA. Association for Computing Machinery.

[Cheryan et al., 2009] Cheryan, S., Plaut, V. C., Davies, P. G., and Steele, C. M. (2009). Ambient belonging: how stereotypical cues impact gender participation in computer science. *Journal of personality and social psychology*, 97(6):1045–1060. Place: United States.

[Chiodini et al., 2021] Chiodini, L., Moreno Santos, I., Gallidabino, A., Tafliovich, A., Santos, A. L., and Hauswirth, M. (2021). A curated inventory of programming language misconceptions. In *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE '21, page 380–386, New York, NY, USA. Association for Computing Machinery.

[Christensen et al., 2021] Christensen, I. M., Marcher, M. H., Grabarczyk, P., Graversen, T., and Brabrand, C. (2021). Computing educational activities involving people rather than things appeal more to women (recruitment perspective). In *Proceedings of the 17th ACM Conference on International Computing Education Research*, ICER 2021, page 127–144, New York, NY, USA. Association for Computing Machinery.

[Clear et al., 2008] Clear, T., Whalley, J., Lister, R., Carbone, A., Hu, M., Sheard, J., Simon, B., and Thompson, E. (2008). Reliably classifying novice programmer exam responses using the solo taxonomy. In Mann, S. and Lopez, M., editors, *Proceedings of the Twenty First Annual Conference of the National Advisory Committee on Computing Qualifications*, pages 23 – 30. National Advisory Committee on Computing Qualifications (NACCQ). 21st National Advisory Committee on Computing Qualifications, NACCQ 2008 ; Conference date: 04-07-2008 Through 07-07-2008.

[Corbin and Strauss, 2008] Corbin, J. and Strauss, A. (2008). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. SAGE Publications, Inc., Thousand Oaks, California, 3rd edition.

[Corney et al., 2014] Corney, M., Fitzgerald, S., Hanks, B., Lister, R., McCauley, R., and Murphy, L. (2014). 'explain in plain english' questions revisited: Data structures problems. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, SIGCSE '14, page 591–596, New York, NY, USA. Association for Computing Machinery.

[Corney et al., 2011] Corney, M., Lister, R., and Teague, D. (2011). Early relational reasoning and the novice programmer: Swapping as the "hello world" of relational reasoning. In *Proceedings of the Thirteenth Australasian Computing Education Conference - Volume 114*, ACE '11, page 95–104, AUS. Australian Computer Society, Inc.

[Corney et al., 2012] Corney, M., Teague, D., Ahadi, A., and Lister, R. (2012). Some empirical results for neo-piagetian reasoning in novice programmers and the relationship to code explanation questions. In *Proceedings of the Fourteenth Australasian Computing Education Conference - Volume 123*, ACE '12, page 77–86, AUS. Australian Computer Society, Inc.

[De Oliveira Neto and Dobslaw, 2024] De Oliveira Neto, F. G. and Dobslaw, F. (2024). Building collaborative learning: Exploring social annotation in introductory programming. In *2024 IEEE/ACM 46th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, pages 12–21.

[Deissenboeck and Pizka, 2006] Deissenboeck, F. and Pizka, M. (2006). Concise and consistent naming. *Software Quality Journal*, 14(3):261–282.

[Doukakis et al., 2007] Doukakis, D., Grigoriadou, M., and Tsaganou, G. (2007). Understanding the programming variable concept with animated interactive analogies. In *Proceedings of the 8th Hellenic European Research on Computer Mathematics & its Applications Conference, HERCMA'07*.

[Endres et al., 2021a] Endres, M., Fansher, M., Shah, P., and Weimer, W. (2021a). To read or to rotate? comparing the effects of technical reading training and spatial skills training on novice programming ability. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2021, page 754–766, New York, NY, USA. Association for Computing Machinery.

[Endres et al., 2021b] Endres, M., Karas, Z., Hu, X., Kovelman, I., and Weimer, W. (2021b). Relating reading, visualization, and coding for new programmers: A neuroimaging study. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pages 600–612, Madrid, ES. IEEE.

[Ennis, 2018] Ennis, R. H. (2018). Critical Thinking Across the Curriculum: A Vision. *Topoi*, 37(1):165–184.

[Fakhoury et al., 2020] Fakhoury, S., Roy, D., Ma, Y., Arnaoudova, V., and Adesope, O. (2020). Measuring the impact of lexical and structural inconsistencies on developers' cognitive load during bug localization. *Empirical Software Engineering*, 25(3):2140–2178.

[Feitelson, 2023] Feitelson, D. G. (2023). From code complexity metrics to program comprehension. *Commun. ACM*, 66(5):52–61.

[Feitelson et al., 2022] Feitelson, D. G., Mizrahi, A., Noy, N., Shabat, A. B., Eliyahu, O., and Sheffer, R. (2022). How developers choose names. *IEEE Transactions on Software Engineering*, 48(01):37–52.

[Fincher et al., 2020] Fincher, S., Jeuring, J., Miller, C. S., Donaldson, P., du Boulay, B., Hauswirth, M., Hellas, A., Hermans, F., Lewis, C., Mühling, A., Pearce, J. L., and Petersen, A. (2020). Notional machines in computing education: The education of attention. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education*, ITiCSE-WGR '20, page 21–50, New York, NY, USA. Association for Computing Machinery.

[Fincher and Robins, 2019] Fincher, S. A. and Robins, A. V., editors (2019). *The Cambridge Handbook of Computing Education Research*. Cambridge Handbooks in Psychology. Cambridge University Press.

[Floyd et al., 2017] Floyd, B., Santander, T., and Weimer, W. (2017). Decoding the representation of code in the brain: An fmri study of code review and expertise. In *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pages 175–186, Buenos Aires, Argentina. IEEE.

[Fowler et al., 2021] Fowler, M., Chen, B., Azad, S., West, M., and Zilles, C. (2021). Autograding "explain in plain english" questions using NLP. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, page 1163–1169, New York, NY, USA. Association for Computing Machinery.

[Gellenbeck and Cook, 1991] Gellenbeck, E. M. and Cook, C. R. (1991). An investigation of procedure and variable names as beacons during program comprehension. Technical report, Origon State University, USA.

[Gienow, 2017] Gienow, M. (2017). Code noob: The (variable) naming is the hardest part. https://thenewstack.io/code-noob-naming-hardest-part/

[Glassman et al., 2015] Glassman, E. L., Fischer, L., Scott, J., and Miller, R. C. (2015). Foobaz: Variable name feedback for student code at scale. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, page 609–617, New York, NY, USA. Association for Computing Machinery.

[Gobil et al., 2009] Gobil, A. R. M., Shukor, Z., and Mohtar, I. A. (2009). Novice difficulties in selection structure. In *2009 International Conference on Electrical Engineering and Informatics*, volume 02, pages 351–356.

[Gresta et al., 2021] Gresta, R., Durelli, V., and Cirilo, E. (2021). Naming practices in java projects: An empirical study. In *XX Brazilian Symposium on Software Quality*, SBQS '21, New York, NY, USA. Association for Computing Machinery.

[Grover and Basu, 2017] Grover, S. and Basu, S. (2017). Measuring student learning in introductory block-based programming: Examining misconceptions of loops, variables, and boolean logic. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, page 267–272, New York, NY, USA. Association for Computing Machinery.

[Hermans, 2020] Hermans, F. (2020). Hedy: A gradual language for programming education. In *Proceedings of the 2020 ACM Conference on International Computing Education Research*, ICER '20, page 259–270, New York, NY, USA. Association for Computing Machinery.

[Hermans et al., 2018a] Hermans, F., Swidan, A., and Aivaloglou, E. (2018a). Code phonology: An exploration into the vocalization of code. In *Proceedings of the 26th Conference on Program Comprehension*, ICPC '18, page 308–311, New York, NY, USA. Association for Computing Machinery.

[Hermans et al., 2018b] Hermans, F., Swidan, A., Aivaloglou, E., and Smit, M. (2018b). Thinking out of the box: Comparing metaphors for variables in programming education. In *Proceedings of the 13th Workshop in Primary and Secondary Computing Education*, WiPSCE '18, New York, NY, USA. Association for Computing Machinery.

[Highland, 2019] Highland, M. (2019). *Coding for Kids: Scratch: Learn Coding Skills, Create 10 Fun Games, and Master Scratch*. ROCKRIDGE PR.

[Hofmeister et al., 2017] Hofmeister, J., Siegmund, J., and Holt, D. V. (2017). Shorter identifier names take longer to comprehend. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 217–227.

[Izu et al., 2019] Izu, C., Schulte, C., Aggarwal, A., Cutts, Q., Duran, R., Gutica, M., Heinemann, B., Kraemer, E., Lonati, V., Mirolo, C., and Weeda, R. (2019). Fostering program comprehension in novice programmers - learning activities and learning trajectories. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education*, ITiCSE-WGR '19, page 27–52, New York, NY, USA. Association for Computing Machinery.

[Kaczmarczyk et al., 2010] Kaczmarczyk, L. C., Petrick, E. R., East, J. P., and Herman, G. L. (2010). Identifying student misconceptions of programming. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, SIGCSE '10, page 107–111, New York, NY, USA. Association for Computing Machinery.

[Keller, 1990] Keller, D. (1990). A guide to natural naming. *ACM SIGPLAN Notices*, 25:95–102.

[Kennedy and Kraemer, 2019] Kennedy, C. and Kraemer, E. T. (2019). Qualitative observations of student reasoning: Coding in the wild. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '19, page 224–230, New York, NY, USA. Association for Computing Machinery.

[Kenny and Fourie, 2015] Kenny, M. and Fourie, R. (2015). Contrasting classic, straussian, and constructivist grounded theory: Methodological and philosophical conflicts. *Qualitative Report*, 20(8):1270–1289.

[Keuning et al., 2019] Keuning, H., Heeren, B., and Jeuring, J. (2019). How teachers would help students to improve their code. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '19, page 119–125, New York, NY, USA. Association for Computing Machinery.

[Kim and Wilkinson, 2019] Kim, M.-Y. and Wilkinson, I. A. (2019). What is dialogic teaching? constructing, deconstructing, and reconstructing a pedagogy of classroom talk. *Learning, Culture and Social Interaction*, 21:70–86.

[Kirschner et al., 2006] Kirschner, P. A., Sweller, J., and Clark, R. E. (2006). Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist*, 41(2):75–86.

[Kohn, 2017] Kohn, T. (2017). Variable evaluation: An exploration of novice programmers' understanding and common misconceptions. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, page 345–350, New York, NY, USA. Association for Computing Machinery.

[Koksal, 2020] Koksal, I. (2020). The rise of online learning. https://www.forbes.com/sites/ilkerkoksal/2020/05/02/the-rise-of-online-learning/?sh=28f26e472f3c

[Laakso et al., 2008] Laakso, M.-J., Malmi, L., Korhonen, A., Rajala, T., Kaila, E., and Salakoski, T. (2008). Using roles of variables to enhance novices debugging work. *Journal of Information Technology Education: Innovations in Practice*, 5:281–296.

[Lawrie et al., 2007a] Lawrie, D., Feild, H., and Binkley, D. (2007a). Quantifying identifier quality: an analysis of trends. *Empirical Software Engineering*, 12(4):359–388.

[Lawrie et al., 2006] Lawrie, D., Morrell, C., Feild, H., and Binkley, D. (2006). What's in a name? a study of identifiers. In *14th IEEE International Conference on Program Comprehension (ICPC'06)*, pages 3–12.

[Lawrie et al., 2007b] Lawrie, D., Morrell, C., Feild, H., and Binkley, D. (2007b). Effective identifier names for comprehension and memory. *Innovations in Systems and Software Engineering*, 3(4):303–318.

[Lehtinen et al., 2021a] Lehtinen, T., Lukkarinen, A., and Haaranen, L. (2021a). Students struggle to explain their own program code. In *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE '21, page 206–212, New York, NY, USA. Association for Computing Machinery.

[Lehtinen et al., 2021b] Lehtinen, T., Santos, A. L., and Sorva, J. (2021b). Let's ask students about their programs, automatically. In *2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC)*, pages 467–475. Association for Computing Machinery.

[Lister, 2016] Lister, R. (2016). Toward a developmental epistemology of computer programming. In *ACM International Conference Proceeding Series*, volume 13-15-Octo, pages 5–16. Association for Computing Machinery.

[Lister, 2020] Lister, R. (2020). On the cognitive development of the novice programmer: And the development of a computing education researcher. In *Proceedings of the 9th Computer Science Education Research Conference*, New York, NY, USA. Association for Computing Machinery.

[Lister et al., 2009] Lister, R., Fidge, C., and Teague, D. (2009). Further evidence of a relationship between explaining, tracing and writing skills in introductory programming. In *Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education*, ITiCSE '09, page 161–165, New York, NY, USA. Association for Computing Machinery.

[Lister et al., 2006] Lister, R., Simon, B., Thompson, E., Whalley, J. L., and Prasad, C. (2006). Not seeing the forest for the trees: Novice programmers and the solo taxonomy. In *Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, ITICSE '06, page 118–122, New York, NY, USA. Association for Computing Machinery.

[Lopez et al., 2008] Lopez, M., Whalley, J., Robbins, P., and Lister, R. (2008). Relationships between reading, tracing and writing skills in introductory programming. In *ICER '08 - Proceedings of the ACM Workshop on International Computing Education Research*, pages 101–111. ACM Press.

[Makda and Mamazai, 2022] Makda, U. and Mamazai, T. (2022). *Python Coding for Kids Ages 10+*. Independently published.

[Marcher et al., 2021] Marcher, M. H., Christensen, I. M., Grabarczyk, P., Graversen, T., and Brabrand, C. (2021). Computing educational activities involving people rather than things appeal more to women (cs1 appeal perspective). In *Proceedings of the 17th ACM Conference on International Computing Education Research*, ICER 2021, page 145–156, New York, NY, USA. Association for Computing Machinery.

[Marji, 2014] Marji, M. (2014). *Learn to program with Scratch: A visual introduction to programming with games, art, science, and math*. No Starch Press.

[Mattarella-Micke and Beilock, 2010] Mattarella-Micke, A. and Beilock, S. L. (2010). Situating math word problems: The story matters. *Psychonomic Bulletin & Review*, 17(1):106–111.

[Matthes, 2023] Matthes, E. (2023). *Python Crash Course. A hands-on, project-based introduction to programming (3rd edition)*. No Starch Press.

[McMaster et al., 2016] McMaster, K., Rague, B., Sambasivam, S., and Wolthuis, S. (2016). Coverage of cs1 programming concepts in c++ and java textbooks. In *2016 IEEE Frontiers in Education Conference (FIE)*, page 1–8. IEEE Press.

[McMaster et al., 2018] McMaster, K., Rague, B., Sambasivam, S., and Wothuis, S. (2018). Software concepts emphasized in introductory programming textbooks. In *Proceedings of the 19th Annual SIG Conference on Information Technology Education*, SIGITE '18, page 91, New York, NY, USA. Association for Computing Machinery.

[Morrison et al., 2015] Morrison, B. B., Margulieux, L. E., and Guzdial, M. (2015). Subgoals, context, and worked examples in learning computing problem solving. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, ICER '15, page 21–29, New York, NY, USA. Association for Computing Machinery.

[Murphy et al., 2012a] Murphy, L., Fitzgerald, S., Lister, R., and McCauley, R. (2012a). Ability to 'explain in plain english' linked to proficiency in computer-based programming. In *Proceedings of the Ninth Annual International Conference on International Computing Education Research*, ICER '12, page 111–118, New York, NY, USA. Association for Computing Machinery.

[Murphy et al., 2012b] Murphy, L., McCauley, R., and Fitzgerald, S. (2012b). 'explain in plain english' questions: Implications for teaching. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, SIGCSE '12, page 385–390, New York, NY, USA. Association for Computing Machinery.

[Newman et al., 2020] Newman, C. D., AlSuhaibani, R. S., Decker, M. J., Peruma, A., Kaushik, D., Mkaouer, M. W., and Hill, E. (2020). On the generation, structure, and semantics of grammar patterns in source code identifiers. *Journal of Systems and Software*, 170:110740.

[Pelchen and Lister, 2019] Pelchen, T. and Lister, R. (2019). On the frequency of words used in answers to explain in plain english questions by novice programmers. In *Proceedings of the Twenty-First Australasian Computing Education Conference*, page 11–20, New York, NY, USA. Association for Computing Machinery.

[Peruma et al., 2018] Peruma, A., Mkaouer, M. W., Decker, M. J., and Newman, C. D. (2018). An empirical investigation of how and why developers rename identifiers. In *Proceedings of the 2nd International Workshop on Refactoring*, IWoR 2018, page 26–33, New York, NY, USA. Association for Computing Machinery.

[Prat et al., 2020] Prat, C. S., Madhyastha, T. M., Mottarella, M. J., and Kuo, C.-H. (2020). Relating Natural Language Aptitude to Individual Differences in Learning Programming Languages. *Scientific Reports*, 10(1):3817.

[Rich et al., 2022] Rich, K. M., Franklin, D., Strickland, C., Isaacs, A., and Eatinger, D. (2022). A learning trajectory for variables based in computational thinking literature: Using levels of thinking to develop instruction. *Computer Science Education*, 32(2):213–234.

[Rich et al., 2017] Rich, K. M., Strickland, C., Binkowski, T. A., Moran, C., and Franklin, D. (2017). K-8 learning trajectories derived from research literature: Sequence, repetition, conditionals. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, ICER '17, page 182–190, New York, NY, USA. Association for Computing Machinery.

[Robbins, 2023] Robbins, P. (2023). *Python Programming for Beginners*. Independently published.

[Sajaniemi, 2002] Sajaniemi, J. (2002). An empirical analysis of roles of variables in novice-level procedural programs. In *Proceedings IEEE 2002 Symposia on Human Centric Computing Languages and Environments*, pages 37–39.

[Sajaniemi and Kuittinen, 2005] Sajaniemi, J. and Kuittinen, M. (2005). An experiment on using roles of variables in teaching introductory programming. *Computer Science Education*, 15(1):59–82.

[Sajaniemi and Prieto, 2005] Sajaniemi, J. and Prieto, R. N. (2005). Roles of Variables in Experts ' Programming Knowledge. In Romero, P., Good, J., Acosta Chaparro, E., and Bryant, S., editors, *17th Workshop of the Psychology of Programming Interest Group (PPIG17)*, pages 145–159. Sussex University.

[Salac and Franklin, 2020] Salac, J. and Franklin, D. (2020). If they build it, will they understand it? exploring the relationship between student code and performance. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '20, page 473–479, New York, NY, USA. Association for Computing Machinery.

[Salac et al., 2020] Salac, J., Jin, Q., Klain, Z., Turimella, S., White, M., and Franklin, D. (2020). Patterns in elementary-age student responses to personalized & generic code comprehension questions. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, SIGCSE '20, page 514–520, New York, NY, USA. Association for Computing Machinery.

[Santos and Sousa, 2017] Santos, A. L. and Sousa, H. (2017). An exploratory study of how programming instructors illustrate variables and control flow. In *Proceedings of the 17th Koli Calling International Conference on Computing Education Research*, Koli Calling '17, page 173–177, New York, NY, USA. Association for Computing Machinery.

[Schankin et al., 2018] Schankin, A., Berger, A., Holt, D. V., Hofmeister, J. C., Riedel, T., and Beigl, M. (2018). Descriptive compound identifier names improve source code comprehension. In *Proceedings of the 26th Conference on Program Comprehension*, ICPC '18, page 31–40, New York, NY, USA. Association for Computing Machinery.

[Schley and Fujita, 2014] Schley, D. R. and Fujita, K. (2014). Seeing the math in the story: On how abstraction promotes performance on mathematical word problems. *Social Psychological and Personality Science*, 5(8):953–961.

[Schulte, 2008] Schulte, C. (2008). Block model: An educational model of program comprehension as a tool for a scholarly approach to teaching. In *Proceedings of the Fourth International Workshop on Computing Education Research*, ICER '08, page 149–160, New York, NY, USA. Association for Computing Machinery.

[Sharif and Maletic, 2010] Sharif, B. and Maletic, J. I. (2010). An eye tracking study on camelcase and under_score identifier styles. In *2010 IEEE 18th International Conference on Program Comprehension*, pages 196–205.

[Sheard et al., 2014] Sheard, J., Simon, Dermoudy, J., D'Souza, D., Hu, M., and Parsons, D. (2014). Benchmarking a set of exam questions for introductory programming. In *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*, pages 113–121. Australian Computer Society, Inc.

[Stegeman et al., 2014] Stegeman, M., Barendsen, E., and Smetsers, S. (2014). Towards an empirically validated model for assessment of code quality. In *Proceedings of the 14th Koli Calling International Conference on Computing Education Research*, Koli Calling '14, page 99–108, New York, NY, USA. Association for Computing Machinery.

[Stegeman et al., 2016] Stegeman, M., Barendsen, E., and Smetsers, S. (2016). Designing a rubric for feedback on code quality in programming courses. In *Proceedings of the 16th Koli Calling International Conference on Computing Education Research*, Koli Calling '16, page 160–164, New York, NY, USA. Association for Computing Machinery.

[Sweigart, 2020] Sweigart, A. (2020). *Automate the boring stuff with Python. Practical programming for total beginners (2nd edition)*. No Starch Press.

[Sweller, 2011] Sweller, J. (2011). *Cognitive Load Theory*, volume 55. Academic Press. ISSN: 00797421 Publication Title: Psychology of Learning and Motivation - Advances in Research and Theory.

[Swidan and Hermans, 2019] Swidan, A. and Hermans, F. (2019). The effect of reading code aloud on comprehension: An empirical study with school students. In *Proceedings of the ACM Conference on Global Computing Education*, CompEd '19, page 178–184, New York, NY, USA. Association for Computing Machinery.

[Swidan et al., 2017] Swidan, A., Serebrenik, A., and Hermans, F. (2017). How do scratch programmers name variables and procedures? In *2017 IEEE 17th International Working Conference on Source Code Analysis and Manipulation (SCAM)*, pages 51–60.

[Tacke, 2019] Tacke, A. B. (2019). *Coding for Kids Python*. Rockridge Press.

[Takang et al., 1996] Takang, A. A., Grubb, P. A., and Macredie, R. D. (1996). The effects of comments and identifier names on program comprehensibility: an experimental investigation. *J. Program. Lang.*, 4:143–167.

[Teasley, 1994] Teasley, B. E. (1994). The effects of naming style and expertise on program comprehension. *International Journal of Human-Computer Studies*, 40(5):757–770.

[Tshukudu et al., 2021] Tshukudu, E., Cutts, Q., Goletti, O., Swidan, A., and Hermans, F. (2021). Teachers' views and experiences on teaching second and subsequent programming languages. In *Proceedings of the 17th ACM Conference on International Computing Education Research*, ICER 2021, page 294–305, New York, NY, USA. Association for Computing Machinery.

[van den Aker and Rahimi, 2024] van den Aker, E. and Rahimi, E. (2024). Design principles for generating and presenting automated formative feedback on code quality using software metrics. In *2024 IEEE/ACM 46th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, ICSE-SEET '24, page 139–150, New York, NY, USA. Association for Computing Machinery.

[van der Werf, 2024] van der Werf, V. (2024). Fostering a natural language approach in programming education (doctoral consortium). In *Proceedings of the 23rd Koli Calling International Conference on Computing Education Research*, Koli Calling '23, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2022a] van der Werf, V., Aivaloglou, E., Hermans, F., and Specht, M. (2022a). (how) should variables and their naming be taught in novice programming education? In *Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 2*, ICER '22, page 53–54, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2022b] van der Werf, V., Aivaloglou, E., Hermans, F., and Specht, M. (2022b). What does this python code do? an exploratory analysis of novice students' code explanations. In *Proceedings of the 10th Computer Science Education Research Conference*, CSERC '21, page 94–107, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2024a] van der Werf, V., Hermans, F., Specht, M., and Aivaloglou, E. (2024a). Promoting deliberate naming practices in programming education: A set of interactive educational activities. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 1*, SIGCSE Virtual 2024, page 235–241, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2024b] van der Werf, V., Hermans, F., Specht, M., and Aivaloglou, E. (2024b). Variables and variable naming in popular programming textbooks for children and novices. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 1*, SIGCSE Virtual 2024, page 242–248, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2024c] van der Werf, V., Swidan, A., Hermans, F., Specht, M., and Aivaloglou, E. (2024c). Teachers' beliefs and practices on the naming of variables in introductory python programming courses. In *2024 IEEE/ACM 46th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, ICSE-SEET '24, page 368–379, New York, NY, USA. Association for Computing Machinery.

[van der Werf et al., 2023] van der Werf, V., Zhang, M. Y., Aivaloglou, E., Hermans, F., and Specht, M. (2023). Variables in practice. an observation of teaching variables in introductory programming moocs. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2023, page 208–214, New York, NY, USA. Association for Computing Machinery.

[van Merriënboer and Kirschner, 2013] van Merriënboer, J. J. and Kirschner, P. A. (2013). *Ten steps to complex learning. A systematic approach to four-component instructional design.* Routledge, second edition.

[Veldthuis and Hermans, 2024] Veldthuis, M. and Hermans, F. (2024). A word about programming: Applying a natural language vocabulary acquisition model to programming education. In *Proceedings of the 2024 ACM SIGPLAN International Symposium on SPLASH-E*, SPLASH-E '24, page 56–65, New York, NY, USA. Association for Computing Machinery.

[Venables et al., 2009] Venables, A., Tan, G., and Lister, R. (2009). A closer look at tracing, explaining and code writing skills in the novice programmer. In *ICER'09 - Proceedings of the 2009 ACM Workshop on International Computing Education Research*, pages 117–128. ACM Press.

[Vermeulen et al., 2000] Vermeulen, A., Ambler, S. W., Bumgardner, G., Metz, E., Misfeldt, T., Shur, J., and Thompson, P. (2000). *The Elements of Java Style*. Cambridge University Press.

[Vorderman et al., 2017] Vorderman, C., Steele, C., Quigley, C., Goodfellow, M., McCafferty, D., and Woodcock, J. (2017). *Coding Projects in Python*. DK Publishing.

[Vorderman et al., 2018] Vorderman, C., Steele, C., Quigley, C., McCafferty, D., and Goodfellow, M. (2018). *Coding Games in Python*. DK Publishing.

[Waguespack, 1989] Waguespack, L. J. (1989). Visual metaphors for teaching programming concepts. *SIGCSE Bull.*, 21(1):141–145.

[Wainewright, 2020] Wainewright, M. (2020). *Code Your Own Games!: 20 Games to Create with Scratch*. Union Square Kids.

[Weeda et al., 2020] Weeda, R., Izu, C., Kallia, M., and Barendsen, E. (2020). Towards an assessment rubric for eipe tasks in secondary education: Identifying quality indicators and descriptors. In *Koli Calling '20: Proceedings of the 20th Koli Calling International Conference on Computing Education Research*, Koli Calling '20, New York, NY, USA. Association for Computing Machinery.

[Whalley and Kasto, 2014] Whalley, J. and Kasto, N. (2014). How difficult are novice code writing tasks? a software metrics approach. In *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148*, pages 105–112. Australian Computer Society, Inc.

[Whalley et al., 2006] Whalley, J. L., Lister, R., Thompson, E., Clear, T., Robbins, P., Kumar, P. K. A., and Prasad, C. (2006). An australasian study of reading and comprehension skills

in novice programmers, using the bloom and solo taxonomies. In *Proceedings of the 8th Australasian Conference on Computing Education - Volume 52*, ACE '06, page 243–252, AUS. Australian Computer Society, Inc.

[Woodcock, 2015] Woodcock, J. (2015). *Coding Games in Scratch: A Step-by-Step Visual Guide to Building Your Own Computer Games*. Penguin.

[Woodcock, 2016] Woodcock, J. (2016). *Coding Projects in Scratch: A Step-by-Step Visual Guide to Coding Your Own Animations, Games, Simulations, and More!* Penguin.

[Xia et al., 2018] Xia, X., Bao, L., Lo, D., Xing, Z., Hassan, A. E., and Li, S. (2018). Measuring program comprehension: A large-scale field study with professionals. *IEEE Transactions on Software Engineering*, 44(10):951–976.

[Xie et al., 2019] Xie, B., Loksa, D., Nelson, G. L., Davidson, M. J., Dong, D., Kwik, H., Tan, A. H., Hwa, L., Li, M., and Ko, A. J. (2019). A theory of instruction for introductory programming skills. *Computer Science Education*, 29(2-3):205–253.

Afterword and acknowledgments

Had anyone told my 2019-past self that I was going to pursue a PhD at the Leiden Institute of Advanced Computer Science, I would have never believed them. While my career path has been anything but straightforward, the leap from archaeology to educational sciences felt like a natural one (what's more embedded in society if not education?), and the leap to data analytics was a logical one (what's data if not pieces of information to interpret, like archaeology?). By transitioning to computer science education, I aimed to contribute to the education of the future. Although I am likely the only one walking this unique path, I have crossed paths with many people to whom I owe thanks.

To start, I thank my supervision team. First, I owe my gratitude to *Felienne*, for your passion, drive, and inspiration; for without it, I would not have been here. I still vividly remember our first conversation after seeing one of your animated talks. "Be brave. You never know what a first meeting might lead to." (James Norbury, Big Panda and Tiny Dragon). I would also not have been here without *Fenia*; you always knew like clockwork when I needed a little encouragement. Your 'complaint' of "I want to use my red pen and draw arrows and stuff," may have been the single most helpful comment throughout the entire process. And of course, I am grateful for your warm advice, our rejection drinks, and your home dinners, which made me feel part of an academic family. I also thank *Marcus*, you blindly trusted me to make it happen, and from you I learned to believe in myself. I thank all of you for the opportunity to work at LIACS and CEL, and for bringing together a bunch of 'lost' PhD students.

I extend my gratitude to my colleagues and peers at Leiden and Delft University, with whom I had much-needed coffee breaks, tears, and laughter, and with whom I shared hardship, networks, food, and advice. Thank you for the opportunities you gave me to grow and experiment. Special thanks to (no particular order): *Martine Baars*, for trusting me to teach in your course, which I then shamelessly used to experiment according to my own philosophy: dialogue instead of PowerPoint. For me, that was a powerful milestone. *Gillian Saunders-Smits*, for your unwavering support and advice as a woman in science; and for your academic guidance, introducing me to your network, and supporting and enabling the work of our own. *Gitte*, for being my partner in crime as colleagues and friends. The endless flow of fantastical ideas, your fanaticism, our covid homework sessions, our adventures –on and off conferences–, and so much more, have made my journey memorable. I cannot imagine anymore NOT sharing our joy, sadness, frustration, and academic enthusiasm. I am truly grateful to have you as my friend. *Marcella*, for our immediate connection; you too, I have very quickly considered friends first, colleagues second. You've 'hooked' me into crochet and shown me so much strength, endurance, and enthusiasm. Please continue to show everyone the way to being a beautiful, smart, and ambitious superwoman. *Johannes*, for walking up to a few strangers at the hotel lobby and fully embracing our company, crazy stories, and ambitious ideas; you have made our collaboration an absolute joy, bringing in your pragmatism, enthusiasm, planning, and good questions. I would not hesitate to work together again. *Shirley*, for making it look easy to walk the path you lay out for yourself and not letting anyone get to you. And behind your calm front, there is a ton of creativity, of which I can only hope you keep it. *Xiaoling*, for your unwavering dedication to work and hobbies, and your zest for learning everything that finds itself on your path. *Erna*, for your dry humor and for never skipping lunch. *Nesse*, for our long and enthusiastic talks and walks that reminded me doing a PhD is also fun. *Manuel*, for always being in for social drinks and always knowing the right (AI) tools and prompts, both in work and fun settings (who said they are mutually exclusive?) #DrankmakerijDeKievit.

I furthermore thank the SEFI, ITiCSE, and Koli communities, whose members embraced my 'otherness' and encouraged their PhDs. Special thanks to *Barbara Ericson, Matt Daniels, Nick*

Faulkner, Mark Guzdial, for mentoring us towards confidence in the best Doctoral Consortium I have been, also due to my peer students (*Dimitri, Jesse, Sebastian, Katrin*) and the biggest bag of Kinderbueno. I also thank *Lauri Malmi* for the lovely conversation we had on educational theory; even though I only dared to talk to you at the very end of the evening, you were a great reminder of the beautiful academic discourse that can happen just by meeting someone with a similar passion.

My friends outside academia, you kept reminding me, with or without intention, of life outside the university. A special thanks to those who have made an impact on who I am during my PhD journey; I have so much respect for who you are! *Willemijn*, for being your brave, spontaneous self, for puzzles and tea and talks and drinks; *Dirk*, for always being a challenging opponent in almost any game and truly believing in me as if it is the single most normal thing to do; *Mara*, for your awesome energy, strong and reliable advice, and being someone I always look forward to talking to again and again; *Rosanne*, for making our 15-year old friendship feel unchanged even when we are not in contact; and *Kayley*, for reaching out and being so open and inviting. I also thank *Fenny, Dorien, Cailin & Ian, Rosa, Caitlin, Frederic, Jeff* and *Eva* for your unique contributions to my life these last years.

Finally, I am immensely thankful for my close family, who are always there for me, trusting in me, and giving advice. I have much respect for my mom, *Willeke*, who finished her PhD right before I started mine; you were in many ways my rock, and your often spot-on advice is as good as internalized. Thank you for raising me in a surrounding that exposed me to lots of cultures, interests, and opinions. Likewise, I can always rely on the practical and pragmatic advice from my dad, *Willy*; your voice in my head keeps reminding me to work to live, not the other way around. I am lucky to feel how proud you both are of me. *Marijn* and *Fré*, I think you both are so smart and brave and cool, and even though I wish I could be more like you, I could never do what you do. And finally, *Coen*, now pursuing your own PhD, my 30th would not have been the same without you!

I strongly believe all events are connected, as I can connect each choice I made leading to the next cool thing I got to do. In a more general sense, I am proud to say that I have become a sort of human time machine, able to see and connect with the past (archaeology), the now (education), and the future (computer science). All three have become a huge part of who I am and how I view the world. I owe my gratitude to all of you who contributed to this, including my teachers, mentors, professors, and fellow students along the way.