Abstract

Expectations from academics and the industry, to have students and employees who are independent and capable of quickly writing code to resolve work-related issues, are growing high. However, teaching and learning programming is certainly not easy and very challenging. Literature shows that a lot of work has been done to improve this. Nonetheless, it is evident that little effect of this work has had impact on the actual practice of teaching and learning of Software Development programming skills. This gap has been addressed in this paper to enhance the teaching and learning process of programming to students. Furthermore, teaching programming literature research has been classified into 3 categories; teaching approach, teaching model and teaching tool. As a result, this paper proposes the following objectives to tackle this problem: • Identify what research has found out about how to teach and learn programming and other aspects of Software Development • Investigate how and why this research has not been applied to teaching Software Development. How more use could be made of it to improve teaching? Finally, an Enhanced Teaching Model (ETM) has been proposed, which combines several teaching approaches and models from literature. In addition, this model uses teaching tools to provide goal-focused exercises, assess students’ performance and obtain feedback from the learning community. Last but not least, a discussion about the future work required in order to assess the model and thus
improve it.

References

Enhanced Teaching Model (ETM) for Teaching Programming Languages

Technical Symposium on CS Education. Philadelphia, February. pp. 135–139
Educational Computing Research, 9(3), 355–371
solving: the role of programming environments based on structure editors. In Proceedings
of the National Educational Computing Conference, pp. 231–6
Proceedings of INTERCHI Adjunct proceedings Amsterdam, 24 – 9 April 1993, pp. 29–30
level of cognitive maturity and the types of errors made by students in a computer programming
course. College Teaching Methods and Styles Journal, 3, 31–40
Comput. Small Coll, 17(2), 23-33
SIGCSE Bull., 34(4), 99-100
70-77
means of Extreme Programming, Proceedings of the 25th International conference on Software
Engineering (pp. 586-593). Portland, Oregon: IEEE Computer Society
River, New Jersey: Prentice-Hall
- Williamsons, K., Burstein, F., & McKemmish (2002). The two major traditions of
research. In K. Williamsons (Ed.), Research methods for students, academics and
professionals (2nd ed.). Wagga Wagga, New South Wales: Centre for Information Studies,
Charles Sturt University
Conference Proceedings of the 5th Annual SIGCSE/SIGUE Conference on Innovation and
Technology in Computer Science Education”, ITiCSE 2000, Vol. 32, No. 3, pg. 191-191,
ACM Press
ACM Computing Surveys, 3, 121–141
- Perkins, D. N., Hancock, C., Hobbs, R., Martin, F., & Simmons, R. (1986). Conditions
of learning in novice programmers. Journal of Educational Computing Research, 2, 37–56
Instruction, access, and ability. Educational Psychologist, 20, 191-206
- Perkins, D. N., Schwartz, S., & Simmons, R. (1988). Instructional strategies for the
problems of novice programmers. In R. E. Mayer (Ed.), Teaching and learning computer
programming (pp. 153–178). Hillsdale, NJ: Lawrence Erlbaum
Computer-Based Instruction, 16, 11–19
Enhanced Teaching Model (ETM) for Teaching Programming Languages

- Jenkins, T. (2002). ON THE DIFFICULTY OF LEARNING TO PROGRAM. 3rd Annual LTSN-ICS Conference. University of Ulster, LTSN Centre for Information and Computer Sciences

Index Terms

Computer Science Programming Language
Enhanced Teaching Model (ETM) for Teaching Programming Languages

**Keywords**
Enhanced Teaching Model (ETM)  Intelligent Teaching System (ITS)  Teaching Approach  Teaching Model  Teaching Tool  Software Development.