

# Toward an Inventory of Plans and Patterns, and Their Use to Foster Strategic Knowledge in Programming Novices

Umberto Costantini  
Università degli Studi di Milano  
Milan, Italy  
umberto.costantini@studenti.unimi.it

Violetta Lonati  
Università degli Studi di Milano  
Milan, Italy  
Lab. CINI “Informatica e Scuola”  
lonati@di.unimi.it

Anna Morpurgo  
Università degli Studi di Milano  
Milan, Italy  
Lab. CINI “Informatica e Scuola”  
morpurgo@di.unimi.it

## ABSTRACT

When learning to program, different types of knowledge and skills have to be built: syntactic knowledge, i.e., knowledge of the language features; conceptual knowledge, i.e., knowledge of the semantics of the various constructs; strategic knowledge, i.e. the ability to use syntactic and conceptual knowledge in the most appropriate and effective way to solve programming problems. Strategic knowledge also includes the ability to exploit and combine solutions to known problems to solve new ones and is one of the hardest skills to acquire when learning to program [5, 6, 11]. Terms such as goals/plans [10], patterns [2, 7, 8], schema [9], strategies [4], and variables’ roles [3] have been used to refer to this kind of knowledge. Several authors suggest that strategic knowledge should be taught explicitly, however there are three main issues to address:

- (1) plans/patterns/schema are used sometimes interchangeably and sometimes similarly but not correspondingly; it is claimed that a small number of plans/patterns is needed in an introductory programming course, however there is no agreed upon inventory of plans/patterns among the CS community;
- (2) the notion of variables’ roles isn’t always explicitly linked to the notion of plans/patterns;
- (3) little instructional material is available that explicitly teaches strategies, and it is neither easy nor obvious how to design such material.

The aim of this study is to investigate whether and how such notions of plans/patterns/schema and variables’ roles can be integrated into introductory programming courses.

The idea is to establish a framework that includes and arranges the different definitions presented in literature. The framework developed will be used for building a catalogue of plans/pattern and variables’ roles encountered in CS1 practice, in CS1 assessment, and in the literature on introductory programming; this catalogue will be the basis for designing new instructional/learning material for fostering strategic knowledge. The developed material will be used in one of our introductory programming courses and in peer tutoring activities for the students. The effectiveness will be tested

evaluating students programming skills with a pre-test and a post-test, using a group as a treatment group and another one as control group.

## CCS CONCEPTS

• **Social and professional topics** → CS1.

## KEYWORDS

Programming learning; strategic knowledge; CS1; novice programmers; programming plans; programming patterns

## ACM Reference Format:

Umberto Costantini, Violetta Lonati, and Anna Morpurgo. 2020. Toward an Inventory of Plans and Patterns, and Their Use to Foster Strategic Knowledge in Programming Novices. In *Proceedings of the 2020 International Computing Education Research Conference (ICER '20), August 10–12, 2020, Virtual Event, New Zealand*. ACM, New York, NY, USA, 1 page. <https://doi.org/10.1145/3372782.3408115>

## REFERENCES

- [1] Piraye Bayman and Richard E Mayer. 1988. Using conceptual models to teach BASIC computer programming. *Journal of Educational Psychology* 80, 3 (1988), 291.
- [2] Walter Beck, S. Rebecca Thomas, Janet Drake, J. Philip East, and Eugene Wallingford. 1996. Pattern Based Programming Instruction. In *1996 Annual Conference. ASEE Conferences*, Washington, District of Columbia. <https://peer.asee.org/6228>.
- [3] Pauli Byckling and Jorma Sajaniemi. 2006. Roles of variables and programming skills improvement. *ACM SIGCSE Bulletin* 38, 1 (2006), 413–417.
- [4] Michael de Raadt, Richard Watson, and Mark Toleman. 2009. Teaching and assessing programming strategies explicitly. *Conferences in Research and Practice in Information Technology Series* 95 (2009), 45–54.
- [5] David Ginat, Eti Menashe, and Amal Taya. 2013. Novice difficulties with interleaved pattern composition. In *Informatics in Schools. Sustainable Informatics Education for Pupils of all Ages - 6th ISSEP 2013, Oldenburg, Germany, Proceedings*, Vol. 7780 LNCS. Springer, 57–67. [https://doi.org/10.1007/978-3-642-36617-8\\_5](https://doi.org/10.1007/978-3-642-36617-8_5)
- [6] Michael McCracken, Tadeusz Wilusz, Vicki Almstrum, Danny Diaz, Mark Guzdial, Dianne Hagan, Yifat Ben-David Kolikant, Cary Laxer, Lynda Thomas, and Ian Utting. 2001. A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. *ACM SIGCSE Bulletin* 33, 4 (dec 2001), 125. <https://doi.org/10.1145/572139.572181>
- [7] Orna Muller. 2005. Pattern oriented instruction and the enhancement of analogical reasoning. *ICER 2005* (2005), 57–67. <https://doi.org/10.1145/1089786.1089792>
- [8] Viera K. Proulx. 2004. Programming patterns and design patterns in the introductory computer science course. *ACM SIGCSE Bulletin* 32, 1 (jul 2004), 80–84. <https://doi.org/10.1145/331795.331819>
- [9] Robert S. Rist. 1989. Schema creation in programming. *Cognitive Science* 13, 3 (1989), 389–414. [https://doi.org/10.1016/0364-0213\(89\)90018-9](https://doi.org/10.1016/0364-0213(89)90018-9)
- [10] Elliot Soloway. 1986. Learning to Program = Learning to Construct Mechanisms and Explanations. *Communication of the ACM* 29, 9 (1986), 850–858.
- [11] Jacqueline L. Whalley, Raymond Lister, Errol Thompson, Tony Clear, Phil Robbins, P. K. Ajith Kumar, and Christine Prasad. 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* (Hobart, Australia) (ACE '06). Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 243–252. <http://dl.acm.org/citation.cfm?id=1151869.1151901>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ICER '20, August 10–12, 2020, Virtual Event, New Zealand

© 2020 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-7092-9/20/08.

<https://doi.org/10.1145/3372782.3408115>