The Seventh International Conference on eLearning (eLearning-2016), 29 – 30 September 2016, Belgrade, Serbia

OVERVIEW OF ONLINE TEACHING RESOURCES FOR LOGIC PROGRAMMING

SANDRA LOVRENČIĆ
University of Zagreb, Faculty of Organization and Informatics Varaždin, sandra.lovrencic@foi.hr

MIRKO ČUBRILO
University of Zagreb, Faculty of Organization and Informatics Varaždin, mirko.cubrilo@foi.hr

Abstract: *Computer science studies offer a number of programming courses. Previous research, mostly concentrated on object-oriented programming, has shown that students face various obstacles in learning programming languages, which resulted in suggestions for improving teaching process in this field. Logic programming, belonging to a different, declarative programming paradigm, has been researched to a smaller extent, although it raises even bigger challenges for both students and teachers. Students are recommended to first acquire knowledge in formal logic, and then to learn how to apply that knowledge in logic programming. This paper addresses teaching formal logic and especially logic programming. As a first step in initiative for its improvement, an overview of existing logic programming teaching resources is made, with the emphasis on freely available online courses, materials and tools.*

Keywords: *logic programming, formal logic, teaching resources, online*

1. INTRODUCTION

In a number of programming courses in computer science studies students have opportunity to learn various programming languages belonging to different programming paradigms. Most of those languages belong to object-oriented programming (OOP) paradigm or support multiple paradigms, one of them often being OOP. Logic programming languages that belong to declarative programming paradigm are also an essential part of computer science studies, especially those oriented towards artificial intelligence or databases.

Problems connected both to teaching and learning programming in general and especially for OOP or multiple paradigm languages have been researched extensively for decades, which resulted in various teaching and learning proposals and freely available online resources. Logic programming (LP) courses are not that numerous and hence problems in teaching and learning this paradigm have not been researched to sucha an extent. Therefore, the goal of this paper is to present an overview of currently available free online resources that will show possibilites for improvements in the process of teaching and learning LP.

This paper is organized as follows. Chapter 2 explains the importance of formal logic in everyday life and gives an example of teaching formal logic and logical reasoning based in virtual environment. In Chapter 3 we discuss problems and available resources for programming education in general and LP and in Chapter 4 we give an overview of availability of various online resources for LP divided into four groups: specialized educational websites, course resources at educational institutions, logic programming tools and other resources. We conclude the paper in Chapter 5.

2. TEACHING FORMAL LOGIC

We don't claim that knowledge of formal logic is necessary to successfully solve problems within the context of the logic programming paradigm, nor do we claim that such knowledge is sufficient - but we do claim that it is, without doubt, desirable. Independent of us being conscious of this, our cognitive abilities after a certain age contain not only elements of reasoning in first order theories, with first order logic as a basic system of logic, but elements of higher order logic as well, which includes reasoning about knowledge, reasoning about reasoning about knowledge… These aspects of logical reasoning are the subject of the theory of multi-agent systems, very well and very comprehensively presented in [1].

An example of everyday behaviour in which the participants, if even unconsciously (implicitly), reason in the logic of the second and higher orders, is street crossing behaviour related to traffic lights. In short, the situation looks like this: The driver and the pedestrian know the meaning of the traffic lights, but that in itself doesn't suffice to guarantee safe traffic. When e.g. the pedestrian wouldn't know that the driver is aware of the meaning of the traffic lights, he possibly wouldn't dare to cross the street crossing when the green light tells him it is safe to do so. Or, when the driver would know that the pedestrian doesn't know that he knows the meaning of traffic lights, he would try to signalize this in a way, which could be interpreted by the pedestrian in an intended or in an unintended way, which would again have further consequences for the behaviour of the participants.

Teaching exact sciences such as mathematics is often done using “examples”. These “examples” are in their essence models of abstract theories in which variables of different types are substituted with their “values” (to be more

precise, their interpretations or models). As these "examples", in difference to the theories themselves, have "content" that is conceptually closer to students' understanding, "examples" can (and the practice shows they very often do) considerably facilitate conceptual acquiring of corresponding theories.

When we are talking about teaching formal logic, then, if we were to apply principles we presented above, it would be logical to pursue it with some of the formal logic system models as its object domain (considering the fact that formal logic systems have very well developed model theory). Here we will mention the Tarski's World environment, named by the famous Polish mathematician and logician Alfred Tarski.

The problem environment consists of a "chessboard" onto which regularly [1]shaped geometrical objects are placed (Image 1). The meaning of the predicates used to build formulas is obvious from their names [2], as is seen in Image 2, and they are applied in the same manner in logic programming.

Teaching formal logic is done through interaction with computer during which the initial assumption of the player (student) is verified in relation to the truthfulness of the formulas presenting statements (claims) about a given model. Under the assumption that the initial claim contained in the formula put by the player was false, her opponent (the computer) will unmistakably show it to be so. The player will then learn to reason better, based on the experience of her mistakes. If the opposite is the case, the computer will try to "deflect" the player from logically valid reasoning and again point out to the errors.

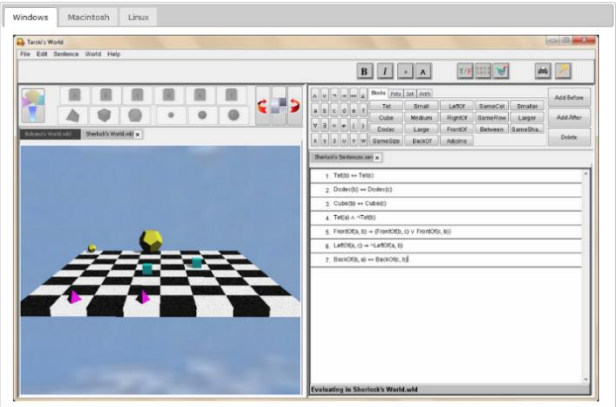

Image 1: The graphical interface of the Tarski's World [2]

A new advance in the same paradigm of teaching formal logic is apparent in the implementation of theories of logical reasoning on temporal-spatial structures, developed in the work of G. Ligozat [3], P. Cabalar i P. E. Santos [4] and other authors. A developed formal frame (a theory) for reasoning on temporal-spatial structures and relations enables the formalization and solving of logical puzzles.

TABLE 2 Blocks language predicates.

Atomic Sentence	Interpretation
Tet(a)	a is a tetrahedron
Cube(a)	a is a cube
Dodec(a)	a is a dodecahedron
Small(a)	a is small
Medium(a)	a is medium
Large(a)	a is large
SameSize(a, b)	a is the same size as b
SameShape(a, b)	a is the same shape as b
Larger(a, b)	a is larger than b
Smaller(a, b)	a is smaller than b
SameCol(a, b)	a is in the same column as b
SameRow(a, b)	a is in the same row as b
Adjoins(a, b)	a and b are located on adjacent (but not diagonally) squares
LeftOf(a, b)	a is located nearer to the left edge of the grid than b
RightOf(a, b)	a is located nearer to the right edge of the grid than b
FrontOf(a, b)	a is located nearer to the front of the grid than b
BackOf(a, b)	a is located nearer to the back of the grid than b
Between(a, b, c)	a, b and c are in the same row, column, or diagonal, and a is between b and c

Image 2: Predicates of the Tarski's World environment[3]

3. PROGRAMMING EDUCATION

Since programming is a very popular and promising field of study, there has been a lot of research regarding its teaching and learning with various proposals of support to this process. Here we shall give just a short overview of this research regarding problems as well as improvement proposals for teaching and learning programming in general and logic programming specifically.

Programming education in general

As in other fields of study, students face various obstacles when learning programming languages. There has been a substantial research regarding those obstacles both for teaching and learning. Three decades ago, difficulties in this process were emphasized in [5] with an overview of new ideas to overcome them. New research shows that those difficulties still haven't been solved [6]. Large research with over 500 students and teachers regarding course contents and course materials has shown that most difficulties refer to the practical parts of courses [7] and other research confirmed the importance of problem solving knowledge [8], examples and practical work [9]. OOP and procedural programming [10] or only OOP [11] were for the most the focus of the research. A number of authors propose visualisation [9][10][11] as a possible method for improving of teaching and learning process in programming education. A short overview of types of auxiliary tools can be found in [12] and an extensive review of research on the subject in [13].

[1] https://ggweb.gradegrinder.net/tarskisworld
[2] Source: https://ggweb.gradegrinder.net/tarskisworld/features;jsessionid=47CE24A1A8A866CA388A51BE814CB798

[3] Source: D.Barker-Plummer, J. Barwise and J. Etchemendy, "Tarski's World: Revised and Expanded", Stanford: Center for the Study of Language and Inf, 2007, pp. 92.

Considering online resources for programming, there are various websites with mostly interactive programming lessons, some freely available and some with monthly subscription, for example Codecademy[4], Treehouse[5], Coursera[6], edX[7] or MIT OpenCourseWare[8]. All of them offer lessons for several or more programming languages mostly aimed at web or database development or supporting multiple programming paradigms. Of course, many universities have openly available materials for their courses online. Hence both tool and online support for programming education in general is substantial.

Logic programming education

According to our experience in teaching logic programming, students in those courses face even bigger obstacles, because methods of solving problems are different than in other programming paradigms that they usually learn earlier, although early research also shows the problem in understanding how such system operates when LP is chosen as introductory course in high school [14]. Difficulties with examples used and reasoning in learning logic programming have been identified with Prolog programming language as the example [15] as well as issues with problem solving [16].

Research has shown that various approaches to teaching LP are used [17]. There has also been a number of suggestions how to improve teaching and learning of logic programming, for example (historically):

- to clearly address misconceptions caused by prior experience with different programming paradigms [18]
- to create a competitive environment [19]
- to put in focus algorithm development [16]
- to use drawing an analogy between LP and natural language argumentation texts within linguistics studies [20]

Tools developed for teaching purposes include competition framework based on Ataxx game [19], MeLoISE platform that uses specific scenario tasks in a collaborative visual interface [21] and online development environment specifically for Answer Set Programming [22].

Online resources are also less available than for programming in general. For example, none of above mentioned websites doesn't offer lessons for logic programming. Situation is somewhat better with freely available online materials for university courses. In general, LP has been researched to a lesser extent and much less tool and online support for teaching and learning process is available.

4. ONLINE TEACHING RESOURCES FOR LOGIC PROGRAMMING

Overview of online teaching resources for logic programming represents initial research about available online materials for introductory and general courses and it includes specialized educational websites, course resources at educational institutions, logic programming tools and other resources. The goal was to find where and to what extent online resources for LP can be found.

Specialized educational websites

There are many educational websites today that offer courses on one or more various subjects. For logic programming subject we examined well known general providers of Massive Open Online Courses (MOOCs) and websites offering programming education specifically.

MOOC providers offer courses and specializations for various subjects from different universities and organizations for free or for a fee. Most known and popular MOOC providers are Coursera, edX, Udacity[9], FutureLearn[10], OpenClassrooms[11], OpenLearning[12], Open2Study[13] and KhanAcademy[14]. We examined them with keywords: "logic programming", "logic" (in case logic programming didn't offer any results) and "Prolog" as a representative programming language. Although analysis shows that programming in general is a subject that has more courses than any other[15], there were no courses as a result for the searched keywords, with the exception of just one course offered by Coursera titled "Introduction to Logic".

Websites that offer only programming education include: Codecademy, Treehouse, Code Avengers[16], Code School[17], CodeHS[18], freeCodeCamp[19], Landofcode[20], and PVT[21]. They offer no courses connected to LP, logic or Prolog, which can also be seen in the comparison of several online resources for learning programming[22].

Course materials at educational institutions

Various educational institutions offer logic programming courses. Some of them have options for free or payed access to online courses or their materials (with or without certificates and credits) and others just publish online materials for regular courses.

Several well-known universities offer online courses or their materials. MIT is the most known for publishing their course materials online and making them freely available

[4] https://www.codecademy.com/
[5] https://teamtreehouse.com/
[6] https://www.coursera.org/
[7] https://www.edx.org/
[8] http://ocw.mit.edu/index.htm
[9] https://www.udacity.com/
[10] https://www.futurelearn.com/
[11] https://openclassrooms.com/
[12] https://www.openlearning.com/
[13] https://www.open2study.com/
[14] https://www.khanacademy.org/
[15] http://www.onlinecoursereport.com/state-of-the-mooc-2016-a-year-of-massive-landscape-change-for-massive-open-online-courses/
[16] https://www.codeavengers.com/
[17] https://www.codeschool.com/
[18] https://codehs.com/
[19] https://www.freecodecamp.com/
[20] http://landofcode.com/
[21] http://pvtuts.com/
[22] http://www.hongkiat.com/blog/sites-to-learn-coding-online/

on MIT OpenCourseWare website. A search showed that LP is just one of topics in available course about programming languages and that also some information about it can be found among formerly used materials for other course. Also, there are three courses about logic and none about Prolog. Materials are available as lecture notes, video or audio lectures, assignments, projects, exams and tools, depending on the course. Stanford[23] offers online courses in the same manner as MOOCs, in our field of interest just two about logic, one of which is also offered at Coursera. Berkley[24] offers course materials up to Spring 2015 (new courses are on edX), but no courses on logic programming. Harvard[25] offers several free online courses and others for a fee, but also none for LP. Yale[26] and UCLA[27] also don't offer any online courses and Carnegie Mellon offers one course on logic[28]. Open University[29], although specialized for distance learning, also doesn't offer any such courses.

Even though there are no online courses for logic programming, many universities offer such courses in their study programs with materials available online. Image 3 shows one of those course pages. Several examples with notes on freely available online materials are:

- The University of Edinburgh[30] - lecture slides, several lecture videos from previous years, programming coursework, tutorials with solutions, old exams, literature and software information and link to online book
- University of Kent[31] - only course description, possibly due to the start of the course this Fall
- York University [32] – only course description, possibly due to the start of the course this Fall, 4-year old lecture slides, assignments, literature and software information[33]
- KTH Royal Institute of Technology[34,35] - two courses (basic and advanced), only course description, links to course pages with no information, possibly due to the start of the courses this Fall
- RWTH Aachen[36] - extensive course script, lecture notes and slides, exercises with solutions, old exams with solutions, literature and software information
- University of Bonn[37] - lecture slides, assignments with solutions, literature and software information
- Technical University of Dresden[38] - lecture slides, tutorials, literature information
- The University of Texas at Dallas[39] - literature and software information

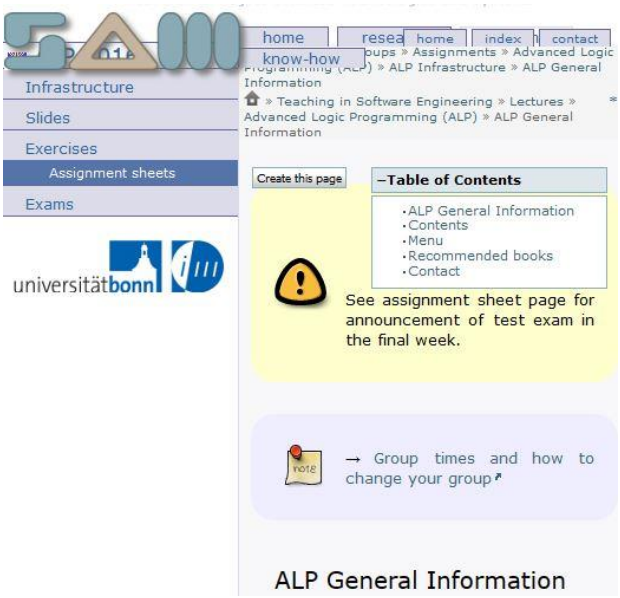

Image 3: LP course page from University of Bonn

Logic Programming Tools

For many courses the main language for teaching logic programming is Prolog, a language that is also specified with two ISO standards: ISO/IEC 13211-1:1995 (general core) and ISO/IEC 13211-2:2000 (modules). There are various tools developed for programming with Prolog that are freely available and open source with extensive tutorials and online community. Mostly used are: SWI-Prolog[40], Visual Prolog[41], Ciao[42], GNU Prolog[43] and XSB[44]. Strawberry Prolog[45] offers a light version for free, and SICStus[46] offers evaluation version.

Other Resources

Various other resources for LP can be found on the Internet, including private web pages and blogs. They vary from simple introduction information[47,48] to web tutorials[49,50] and wiki pages with a large amount of

[23] http://online.stanford.edu/courses
[24] http://webcast.berkeley.edu/
[25] http://www.extension.harvard.edu/academics/online-campus-courses
[26] http://oyc.yale.edu/
[27] https://www.uclaextension.edu/
[28] http://oli.cmu.edu/learn-with-oli/see-our-free-open-courses/
[29] http://www.open.edu/openlearn/free-courses
[30] http://www.inf.ed.ac.uk/teaching/courses/lp/
[31] https://www.kent.ac.uk/courses/modules/module/CO884
[32] http://lassonde.yorku.ca/course-directory/eecs-3401-300
[33] http://www.eecs.yorku.ca/course_archive/2012-13/F/3401/
[34] http://www.kth.se/student/kurser/kurs/ID1213?l=en
[35] http://www.kth.se/student/kurser/kurs/ID2213?l=en
[36] http://verify.rwth-aachen.de/lp15/
[37] https://sewiki.iai.uni-bonn.de/teaching/lectures/alp/2016/start
[38] https://iccl.inf.tu-dresden.de/web/Foundations_of_Logic_Programming_(WS2015)
[39] http://www.utdallas.edu/~gupta/courses/lp/
[40] http://www.swi-prolog.org/
[41] http://www.visual-prolog.com/
[42] http://ciao-lang.org/
[43] http://www.gprolog.org/
[44] http://xsb.sourceforge.net/
[45] http://www.dobrev.com/
[46] https://sicstus.sics.se/
[47] https://bernardopires.com/2013/10/try-logic-programming-a-gentle-introduction-to-prolog/
[48] http://users.isy.liu.se/johanl/yalmip/pmwiki.php?n=Category.LogicProgramming
[49] https://www.cpp.edu/~jrfisher/www/prolog_tutorial/contents.html
[50] http://kti.mff.cuni.cz/~bartak/prolog/

information and links, such as Formal Methods Wiki[51]. Variety of materials can also be found on pages of special interest groups and laboratories, such as Association for Logic Programming[52] and The Computational logic, Languages, Implementation, and Parallelism Laboratory[53].

Most of those information pages and tutorials are mainly oriented towards Prolog. Also, links to various online materials for learning Prolog can be found, among other programming languages, on a GitHub[54] page. One of popular resources is the introductory course Learn Prolog Now![55] that offers an online book, lecture slides and links to Prolog tools, manuals and several other resources.

5. CONCLUSION

Research for freely available online materials for logic programming has shown that the situation is unfavourable when compared to OOP and programming languages belonging to multiple paradigms. Within in the scope of our research, we can conclude the following:

- A short overview of programming resources in general showed that they mostly refer to OOP and multiple paradigm languages and that LP is covered to a smaller extent.
- There are no online courses in logic programming, logic and Prolog on examined MOOCs with the exception of one course on logic, although there are many courses for other programming paradigms.
- There are no available online courses or course materials in LP, logic and Prolog at examined educational institutions, with small exceptions.
- Various educational institutions that offer logic programming courses have course materials freely available online, but they vary greatly from just course descriptions or literature and software information to lecture slides, tutorials and exercises.
- There are many Prolog tools with extensive support available online, considering that it is the main language used in LP education.
- There are various other online resources made available from special groups and laboratories, personal websites and blogs, but mostly oriented towards Prolog.

Therefore, looking at the overall picture, it seems that most of freely available online resources are oriented towards Prolog and accessible from tool websites, unofficial private websites or special group websites. Course materials at educational institutions vary greatly and depend on a lecturer. There are virtually no online logic programming courses either from MOOCs or educational institutions.

According to these results, more effort should be put into development of online courses for LP, as well as into the improvement of resources at websites of educational institutions. Further research will be more oriented towards problems of teaching and learning logic programming with the aim of developing better online resources.

REFERENCES

[1] R. Fagin, J. Y. Halpern, Y. Moses and Moshe Vardi, “Reasoning about knowledge”, Cambridge: MIT Press, 2004, 103-152.

[2] D.Barker-Plummer, J. Barwise and J. Etchemendy, “Tarski's World: Revised and Expanded”, Stanford: Center for the Study of Language and Inf, 2007, pp. 92.

[3] G. Ligozat, “Qualitative Spatial and Temporal Reasoning”, Hoboken: John Wiley & Sons, Inc., 2012.

[4] P. Cabalar and P. E. Santos, “Formalising the Fisherman’s Folly puzzle”, Artificial Intelligence, Vol. 175, pp. 346–377, Jan. 2011.

[5] D. Sleeman, “The challenges of teaching computer programming”, Communications of the ACM, vol. 29, pp. 840-841, Sept. 1986.

[6] M. Konecki, “Problems in Programming Education and Means of Their Improvement”, Chapter 37 in DAAAM International Scientific Book 2014, B. Katalinic, Ed., Vienna: DAAAM International, 2014, pp. 459-470.

[7] E. Lahtinen, K. Ala-Mutka and H.-M. Järvinen, “A study of the difficulties of novice programmers”, in Proc. ITiCSE, 2005, pp. 14-18.

[8] A. Gomes and A.J. Mendes, “Learning to program - difficulties and solutions”, in Proc. ICEE, 2007, http://icee2007.dei.uc.pt/proceedings/papers/411.pdf

[9] S. R. Derus and A. Z. M. Ali, “Difficulties in learning Programming: Views of students”, in Proc. ICIE, 2012, pp. 74–78.

[10] A. Robins, J. Roundtree and N. Roundtree, “Learning and Teaching Programming: A Review and Discussion”, Computer Science Education, Vol. 13, pp. 137-172, April 2003.

[11] I. Milne and G. Rowe, “Difficulties in Learning and Teaching Programming - Views of Students and Tutors”, Education and Information Technologies, Vol. 7, pp. 55-66, March 2002.

[12] M. Konecki, S. Lovrenčić and M. Kaniški, “Using Real Projects as Motivators in Programming Education”, in Proc. MIPRO - Computers in Eucation Conference, 2016, pp. 998- 991.

[13] S. M. Salleha, Z. Shukura and H. M. Judi, “Analysis of Research in Programming Teaching Tools: An Initial Review”, Procedia - Social and Behavioral Sciences, in Proc. IETC, 2013, pp. 127-135.

[14] R. M. Bottino, P. Forcheri and M. T. Molfino, “Teaching computer science through a logic programming approach”, Education and Computing, Vol. 4, pp.71-76, 1988.

[51] http://formalmethods.wikia.com/wiki/Logic_programming
[52] http://www.cs.nmsu.edu/ALP/
[53] https://cliplab.org/
[54] https://github.com/vhf/free-programming-books/blob/master/free-programming-books.md#prolog
[55] http://www.learnprolognow.org/lpnpage.php?pageid=top

[15] Paul Brna, “Logic Programming in Education: a Perspective on the State of the Art”, in Proc. of the Post-Conference Workshop on Logic Programming and Education, 1994, pp. 11-24.

[16] R. L. B. L. Campos, “Logic Programming: Can the learning and teaching process be improved apart from standards adopted by most professors and established in academic books?”, CLEI electronic journal, Vol. 13, Aug. 2010, http://www.clei.org/cleiej/papers/v13i2p2.pdf

[17] S. Yang and M. Joy, “Approaches for Learning Prolog Programming”, Innovation in Teaching and Learning in Information and Computer Sciences, Vol. 6, pp. 88-107, 2007.

[18] D. Stamatis and P. Kefalas, “Logic Programming Didactics”, in Proc. IEEII, 2007, pp. 136-144.

[19] P. Robeiro, H. Simões and M. Ferreira, “Teaching Artificial Intelligence and Logic Programming in a Competitive Environment”, Informatics and Education, Vol. 8, pp. 85-100, April 2009.

[20] N. Ragonis and G. Shilo, “Drawing Analogies between Logic Programming and Natural Language Argumentation Texts to Scaffold Learners’ Understanding”, Journal of Information Technology Education: Research, Vol. 13, pp. 73-89, 2014.

[21] S. Vosinakis, P. Koutsabasis and G. Anastassakis, “A Platform for Teaching Logic Programming Using Virtual Worlds”, in Proc. ICALT'14, pp. 657 - 661, 2014.

[22] C. Reotutar, M. Diagne, E. Balai, E. Wertz, P Lee, S.-L. Yeh and Y. Zhang, “An Online Logic Programming Development Environment”, in Proc. EAAI-16, pp.4130-4131, 2016.