Boosting Conceptual Thinking in a Programming Course – An Ontology-Based Approach

João Paulo Barros
joao.barros@ipbeja.pt
Polytechnic Institute of Beja, Beja, Portugal
Centre of Technology and Systems (UNINOVA-CTS), Caparica, Portugal

ABSTRACT

Students constantly have to learn and relate new concepts. However, those multiple relations are easily not given enough attention as their practical application is often given priority. Hence students are left with incomplete reflection on the subject matter, which quickly implies a surface learning of many essential contents and relations. Here, we propose using simple textual languages for students to express relations between concepts. Students' feedback was highly positive, and they recognized the usefulness of the activity to better reflect and understand multiple intertwining concepts in an object-oriented programming course.

KEYWORDS

Education; pedagogy; active-learning; ontologies; object-oriented programming

ACM Reference Format:
João Paulo Barros. 2023. Boosting Conceptual Thinking in a Programming Course – An Ontology-Based Approach. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2023), March 15–18, 2023, Toronto, ON, Canada.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3545947.3576249

1 INTRODUCTION

Students should understand multiple related concepts to make sense of the underlying complexity around the study matter. To that end, it is necessary to fill gaps between pieces of knowledge, thus creating additional and more profound knowledge that complements the practical application of those concepts.

2 BACKGROUND AND RELATED WORK

Ontologies are formal descriptions of concepts and their mutual relations. Concepts can have associated properties that can be seen as additional related concepts. Ontologies are often exhaustive, quite complex, and around a specific area, and have also been used in education (e.g., [3]). Therefore, the reference tools are complex and do not provide a quick and straightforward way to build even simple ontologies. Here, we report using a simple textual language to make students reflect on concepts and their relationships in the context of an introductory seventy-five hours object-oriented programming course for first-year undergraduate students. We used a web-based environment to readily generate graph representations from the textual specifications.

3 METHODS AND RESULTS

We have used a subset of the DOT language from the Graphviz open-source graph visualization software, allowing the creation of visual representations from text descriptions that students create [1]. We have taken advantage of a free and readily available web-based environment, thus allowing the immediate use of any computer or mobile device [2]. We have tried the approach in class using several ontologies with a distinct focus reflecting the sets of interrelated concepts being studied. Two main approaches were used: (1) prepared ontologies for study and discussion; (2) learning activities where students create ontologies for a given set of concepts or subject areas and later present them to all colleagues and teachers. The former corresponds to already built ontologies given to students; the latter provides a fertile ground for lively in-class discussions about the chosen concepts.

Those ontologies can belong to varied areas and focus on specific contexts, e.g., object-oriented design concepts, programming language concepts, integrated development environment commands, version control systems usage, cyber-physical systems, or even the presentation of assessment elements and rules.

4 CONTRIBUTIONS AND FUTURE WORK

From our experience, this student-centered approach is a valuable tool for self-assessment and learning as it forces students and teachers to formalize their interpretations of a given, small or large, subject area in a precise, simple, visual, and abbreviated way. The creation of ontologies promotes deep learning, and the resulting ontologies add to the study material while also serving as a structured index. Students' feedback was extremely positive. In future work, we intend to apply it to more students and subjects.

ACKNOWLEDGMENTS

This work was funded by Portuguese Agency FCT – Fundação para a Ciência e Tecnologia, in the framework of project UIDB/00066/2020.

REFERENCES

[1] Emden R. Gansner and Stephen C. North. 2000. An open graph visualization system and its applications to software engineering. *Software: Practice and Experience* 30, 11 (2000), 1203–1233. https://doi.org/10.1002/1097-024X(200009)30:11<1203::AID-SPE338>3.0.CO;2-N
[2] GraphvizOnline 2022. Graphviz Online. Available at https://dreampuf.github.io/GraphvizOnline/, accessed on 2022/10/14.
[3] Kristian Stancin, Patrizia Poscic, and Danijela Jaksic. 2020. Ontologies in education – state of the art. *Education and Information Technologies* 25, 6 (01 Nov 2020), 5301–5320. https://doi.org/10.1007/s10639-020-10226-z

SIGCSE 2023, March 15–18, 2023, Toronto, ON, Canada

ACM ISBN 978-1-4503-9433-8/23/03.
https://doi.org/10.1145/3545947.3576249