Easy Coding in Biology: Combining Block-Based Programming Tasks with Biological Education to Encourage Computational Thinking in Girls

Eva Schmidthaler[1][0000−0001−9633−8855],
Corinna Hörmann[1][0000−0002−4770−6217], David Hornsby[1][0000−0003−2080−5706],
Anneliese Fraser[2][0009−0000−9400−0666], Martin Cápay[3][0000−0002−8352−0612], and
Barbara Sabitzer[1][0000−0002−1304−6863]

[1] Johannes Kepler University Linz, Altenbergerstraße 69, 4040 Linz, Austria
eva.schmidthaler@jku.at, corinna.hoermann@jku.at, david.hornsby@jku.at,
barbara.sabitzer@jku.at
[2] University of Passau, Innstraße 41, 94032 Passau, Germany
anneliese.fraser@uni-passau.de
[3] Constantine Philosopher University, Trieda Andreja Hlinku 1, 949 74
Nitra-Chrenová, Slovakia mcapay@ukf.sk

Abstract. This mixed-method research aims to address the gender gap in the Computer Science (CS) field by developing an interdisciplinary STEAM workshop combining practical elements with block-based programming (BBP) tasks on biological topics. The study presents two exploratory workshops called "Easy Coding in Biology" conducted with secondary school students in Slovakia and Austria. The workshops utilized a learning environment called <colette/>, which incorporates block-based coding and augmented reality (AR) features. Forty-seven (female = 23) 11-19-year-old students completed an evaluation questionnaire and observations were made during the workshops to improve the workshop content and process. Preliminary results indicate the potential of combining BBP with biological topics to promote computational thinking (CT) and CS skills in secondary school girls. Some participants faced challenges with the learning environment and programming language, especially younger students, and the use of loops. Adaptations are being made to cater to younger students in science education and include additional CT tasks and experiments. Future courses of the workshop series "Easy Coding in Science" (Physics, Chemistry, and Biology) will be conducted in other countries and with different BBP programs (e.g., MakeyMakey, OzoBlockly, and Micro:bit) in 2023 and 2024.

Keywords: Block-Based Programming · Biology · Coding · Education · Computational Thinking · STEAM

1 Introduction

In 20th-century Europe, women filled roles in mechanics, armament factories, and handicraft businesses due to men's absence during WWII. However, they

J.-P. Pellet and G. Parriaux (Eds.): ISSEP 2023 Local Proceedings, pp. 27–36, 2023.
https://doi.org/10.5281/zenodo.8431893

were later relegated to traditional social positions in the 1950s and 1960s. Interestingly, women predominantly worked as mathematicians for NASA and IBM before computers were developed. The construction of the first calculating machine "ENIAC" in 1946 shifted the focus to male operators. It wasn't until the 1970s that the lack of women in IT became a concern [14]. When looking at female scientists and engineers across EU members, the highest numbers were found in Lithuania (52%) and Bulgaria, Latvia, and Portugal (each 51%), whereas the lowest were discovered in Hungary (33%) and Finland (31%) [7]. In 2021, more than half (54%) of individuals and women (52%) in the EU possessed basic or above basic overall digital skills: proficiency in information and data literacy, communication and collaboration, digital content creation, safety, and problem-solving [7]. Among girls aged 16-19, this percentage significantly increased to 70% [6]. The "Digital Compass" aims for citizens to gain at least 80% basic digital skills by the year 2030. By now, the highest scores of females with basic or above basic digital skills have been found in Malta (98%), Croatia, and Finland (each 93%9, the Czech Republic (89%), and Austria (87%), Slovakia (65%), whereas the lowest shares were registered in Luxembourg (60%), Italy (59%), Bulgaria (51%), and Romania & Germany (each 47%) [6]. In Germany, there are 6.5% fewer first-year students in STEM subjects compared to the last year. But if you take a closer look at the gender distribution and the individual subjects, in the year 2014 over 300,000 German students studied mathematics or natural sciences with 40% female students. Since 2015 the number of students dropped but mathematics and natural sciences students increased to 50% in 2021 [20]; a similar trend can also be observed in Austria, especially in the teaching degree "Biology" [22]. Taking a closer look at Austria, there is still a major difference between male and female STEM students at public universities. In the winter term of 2020/21 there were 143,251 enrolled male students in total at public Austrian universities. Of those, 63,064 (44%) were studying STEM subjects, whereas in contrast only 23,787 (17%) were enrolled in liberal arts. In contrast, the winter term 2020/21 counted 166,815 female students – only 45,987 (28%) were enrolled in STEM majors, while 55,096 (33%) studied liberal arts. Especially when picking out the field of "Computer Science and Communication Technology", there were 18,974 (79%) male and 4,953 (21%) female students in all Austrian public universities. We can even find lower numbers when looking at graduations in 2019/20 in the field of "Computer Science and Communication Technology" (2,203 male, 504 female) [19]. Since girls are still not strongly represented in the STEM field in many European countries [22], [20], there is a high need for teachers and scientists to develop new creative approaches to specifically support young women. Therefore, this work-in-progress aims to show new teaching possibilities to combine BBP and scientific topics from the subject of biology ("Easy Coding in Biology" [17]) in order to promote both disciplines (biology and CS) and to arouse and increase interest in both subjects.

2 Computational Thinking in Science Education

By incorporating CT and CS into biology, students can deepen their understanding of scientific concepts while developing important CT skills applicable across various fields, innovatively and creatively [26], [12]. CT can be taught interdisciplinary in science education through multiple approaches: (1) scientific modeling, where students engage with models to grasp conceptual understanding of phenomena; but still, most teachers teach *about* models, rather *with* them, which includes memorizing models to show conceptual understanding of a certain phenomenon [4] [11] [18]. Another approach is (2) integrating technology by using programming languages, AR/VR applications, educational apps, and data analysis software in science lessons [23], [17]. Additionally, (3) project work and (4) interdisciplinary collaboration, including experiments with STEAM (Science, Technology, Engineering, Arts, Mathematics) teachers, offer further avenues for incorporating CS concepts and supporting CT skills [26], [12].

3 Programming and Science Education

CS and CT are usually not associated with science subjects but student and teacher studies have already demonstrated how to teach and implement CS and CT concepts through workshops and courses in science education in an unplugged or plugged way [13], [3], [9]. Educational applications using BBP, such as MakeyMakey, <colette/>, Scratch, OzoBlockly, or Micro:bit, are promising and possible ways to teach CS skills such as programming. BBP offers shows advantages over text-based programming, making it easier for novice programmers to understand and memorize commands, and reducing syntax errors [28], [27]. In a study comparing block-based and text-based coding, high school students improved their results after attending a coding workshop, with better outcomes and increased interest seen in the BBP group [25]. In 2011, a first attempt was to correct the absence of CT and CS in biological education by introducing a "Computational Biology" course to advanced biology classes, aiming to show high school students how CS is used in the subject of biology and why basic digital skills are necessary for research in many fields of biology [8]. Recent K12 courses have combined text-based programming with biology, such as "Programming in Biology" in 2018 [29] and "Biology Meets Programming: Bio-informatics for Beginners" in 2023 [21]. Block-based coding has been successfully integrated into biology education with tools like "BioBlocks" [10], allowing the programming of biological protocols using visual tools based on Google Blockly and Scratch. Furthermore, the "BioCode" bio-informatics program combines CS and biology by introducing students to CT and programming concepts in the context of biological problems, aiming to enhance students' understanding of biological concepts while developing their programming skills in Python [2]. Another research also showed the successful integration of BBP into STEM subjects like Biology [1]. In a 2020 study, teachers emphasized the importance of BBP in science classes but struggled to design authentic coding activities [23].

4 Methodology

The tasks described in this paper are based on a newly developed mobile Augmented Reality application (mAR) <colette/> (Computational Learning Environment for Teacher in Europe) (available on Android and iOS).

4.1 Research Aim

Women are underrepresented in STEM fields, but the number in natural sciences such as biology is still much higher than in informatics. The "Easy Coding in Biology" workshop series aims to bridge the CS gender gap by blending CS concepts like block-based programming (BBP) with biology education. The participating students are introduced to fundamental concepts such as monocots and dicots, seed germination, and plant biology. This project develops computational thinking (CT) skills among female students while teaching fundamental biology concepts. Through BBP activities, students engage with both subjects, fostering a comprehensive learning experience. Incorporating creativity and interdisciplinary methods, the broader "Easy Coding in Science" initiative addresses various science topics, integrating CS concepts into chemistry, physics, and biology lessons. This ongoing mixed-method pilot study utilizes participatory observation [15], descriptive statistics [24], and Davis' Technology Acceptance Model (TAM)-inspired questionnaires [5]. The goal is to deliver life science content and CT education, focusing on CS competency and gender gap reduction. This report showcases findings from the workshop tasks. The coding exercises were meticulously designed for three-dimensional observation, combining cubes to create biological structures. Each task allowed singular block placement with constant values or sequence assembly via loops and variables. The workshop targets programming skills, CT, and biology understanding. For instance, the pyramid's structure reveals layers, squares, and columns. This systematic analysis led to optimized algorithms through loops and variables, spotlighting advanced solutions as a vital research avenue.

4.2 Experimental Design, Data Collection, and Processing

The first pilot workshop (WB1) "Easy Coding in Biology" from the coding series "Easy Coding in Science" took place in February 2023 in Slovakia [17], at the Constantine Philosopher School in Nitra/Slovakia, and the second workshop (WB2) took place in March 2023 at the Johannes Kepler University in Linz/Austria. In WB1, only the BBP tasks were tested - in Austria also the biological ones. The students were observed by the authors [15], and students' progress, performance, suggestions, and end results were documented. Data collection and processing took place from February-April 2023. The final codes, the answers to the questionnaire, and the log data (number of trials, success rate, use of loops) of the BBP <colette/> learning environment were also analyzed. For the evaluation, a questionnaire (Appendix; Duration=15min) based on Davis' Technology Acceptance Model (TAM) was employed [5]. Descriptive statistics were used to gather and process quantitative data (e.g., opinions on entertainment, difficulty, and interest) [24]. During the workshops, one or two instructors supervised the participants. For assessment, the students used the BBP <colette/> application and its AR feature on their mobile devices. The workshop is divided into four phases: Introduction, Task Assessment, Discussion, and Evaluation (Appendix). The didactic approach, based on Sabitzer's COOL informatics concept, combines hands-on Biology and CS activities, individuality, discovery, cooperation, group discussions, and CT and problem-solving tasks [16]. The project's success is measured by the student's ability to grasp the biology and CS concepts covered during the workshop, achieving this by employing formative assessments via the app (correctness of the code), guided exploration, and application-oriented tasks, which are designed to ensure students' active engagement and attainment of the learning goals.

4.3 Sampling

The study included 47 students from Austria and Slovakia with prior programming experience in Scratch and Python. In February 2023, 37 Slovak secondary school students (19 females), aged 16-19, completed Workshop 1 (WB1) and the 15-minute evaluation questionnaire. Workshop 1 lasted 45 minutes. In Austria, Workshop 2 (WB2) took place in March 2023 with ten gifted students aged 11-12 (four females) from diverse secondary schools. WB2 lasted 35 minutes, followed by a 15-minute evaluation and a 30-minute craft session.

5 Preliminary Results

According to the log data, it took one pair or one student working alone 3.6 trials to successfully complete one of the tasks. Furthermore, the majority of the students in Slovakia and Austria successfully completed the tasks (75%), whereas 50% were using loops. Quantitative data showed that the majority of the participants stated neutral, agreed, or strongly agreed that they liked to collaborate in their regular biology class and in the coding workshop with biological tasks. Around 48% of the female students stated, that they collaborated even more in the coding workshop than in their regular biology class. Twenty-six percent of the girls found the workshop and tasks motivating, and over 56% interesting and/or entertaining because the app is "fun to use". The female participants also stated that 60% had no issues with the BBP app and tasks, while 74% had no problems with the task introduction because they were "easy to understand" (Figures and Preliminary Results, Appendix). Regarding the observations, according to the instructor during WB1, students demonstrated proficiency in completing individual biologic BBP tasks and enjoyed collaborative coding workshops. However, the size of AR markers on the CT scans caused difficulty in task three (Tree), as students had to hold the scan far from their devices to utilize the AR view. Frustration arose when codes were unintentionally deleted due to app switches or crashes. Students were observed debugging their code and incorporating patterns or loops from other codes. Spatial orientation and variable placement posed challenges in using loops. The instructor noted that students felt stressed and required more time for task assessment [17]. In WB2, participants were engaged in tasks despite initial technical difficulties. QR code reading and AR scan issues caused frustration and disappointment again. Still, participants displayed motivation and familiarity with the programming interface. No notable gender differences were observed in BBP, performance, or workshop design. Loop usage varied, with some students mentioning them but not utilizing them unless prompted. The advanced task (Egg) proved challenging for younger students, even those with programming experience. Technical issues persisted with the learning environment (app) and AR feature. The evaluation questionnaire was completed immediately after programming, not after the crafting session.

6 Discussion

In February 2023, during the pilot phase, distinct URLs were assigned to tasks for description access. However, from July 2023, the mobile <colette/> app brought a significant change to task-solving. It offers structured task progression with specific path codes for accessing tasks. The app also allows direct comparison of student solutions with the apps' sample solutions, enhancing students' engagement. It now accommodates various task formats like Parsons puzzles and error identification. Educators can establish digital classroom sessions to monitor student attempts and submissions, refining pedagogical strategies. Findings suggest comparing higher and lower-gifted students may not be appropriate due to their varied abilities. Individual skills and capabilities should be considered when evaluating task performance. After the first workshop, tasks were modified for younger students, with haptic elements for spatial imagination and creativity. Bias potential exists due to the timing of the questionnaire in Austria, warranting adjusted testing for more accurate insights. BBP introduction via Scratch and prior coding experience could lead to coding difficulties due to language transition. Encouraging loop use and emphasizing their significance could mitigate this. Future studies might explore technical alternatives for smoother learning and clearer loop instruction. Time management adjustments were made based on participants' struggles during the workshop. These findings underscore the need to consider individual abilities, address biases, and refine instruction for effective BBP workshop outcomes.

7 Conclusion and Outlook

Important insights were gained from the two workshops, informing the future implementation of "Easy Coding in Biology" workshops across Europe, as for example the time management and task assessment were modified. Tasks were further developed to cater to younger students, and haptic task parts were introduced to enhance spatial imagination and creativity. Overall, the interdisciplinary tasks using the BBP app <colette/> proved interesting for secondary school students. However, technical issues with the learning environment, such as AR markers and scans, need further improvement. Despite these challenges, the majority of participants were able to complete the tasks, use loops and engage with the process. Collaboration in regular biology classes did not show significant changes compared to the workshop. The students in the Austrian and Slovakian pilot study had fun, worked together, and were interested and motivated during the workshop, especially in combination with the creative craft part. Based on the first two workshops, webinars for the teachers will be offered in advance to standardize the timing (evaluation after the workshop). In addition, the teachers involved are interviewed during the online training and the interview is recorded. For the introduction to BBP, the commands of the program were printed out and set up like a puzzle system to facilitate the introduction of the blocks. Further workshops in the subjects of physics and chemistry will follow in 2023 and 2024. More than 200 pupils are expected to take part in the “Easy Coding in Science” workshop series in Germany, Austria, and Slovakia. Consideration is also being given to extending the series to text-based programming.

8 Acknowledgement

This research was partially funded by Slovak Research and Development Agency, grant number APVV-20-0599, by the Ministry of Education, Science, Research and Sport of the Slovak Republic, grant number 015UKF-4/2021.

A Appendix

A.1 Preliminary Results and Course of the Workshop

The workshop “Easy Coding in Biology” is divided into four phases:

1. Introduction: Discussion and introduction of the biological content, task design, the app, and the BBP languages; What is the difference between monocots and dicots (examples of representatives); What does it take to germinate seeds, and how fast do cress seeds germinate? What ingredients does cress have? How to build a plant bed with block-based programming? What commands are required and how can you cleverly shorten the code and incorporate loops?
2. Task Assessment 1 and 2: Solving biological (WB2), crafting (WB2), and CS tasks of the biological items (WB1-2)
3. Discussion: Discussing the issues, benefits, final approaches, problems with the tasks, BBP, and app (WB1-2)
4. Evaluation: Filling out the questionnaire (WB1-2)

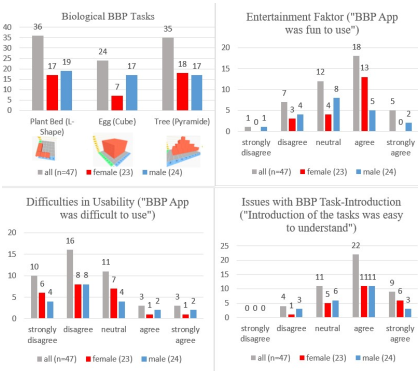

Fig. 1. Number of tasks attempted by students (upper left), students' perceptions on the entertainment factor (upper right), usability (bottom left), and task introduction (bottom right)

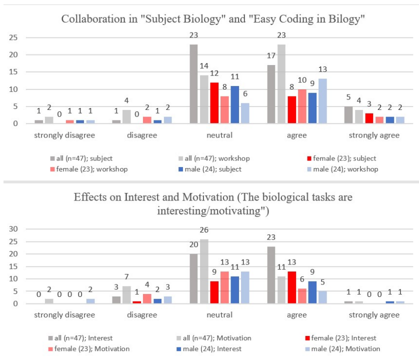

Fig. 2. Students' perceptions on collaboration in regular biology class and in the workshop (above), and on the effects regarding interest and motivation (below)

A.2 Questionnaire

1. Age (open-ended question)
2. Gender (Select one: female, male, non-binary, no gender, no answer)
3. Which task(s) did you work on? Please tick the appropriate answer(s)
   – Create an Algorithm for an Egg
   – Create an Algorithm for a L-Shaped
   – Create an Algorithm for a Conifer
4. It is enjoyable to collaborate in my regular Biology (Science) classes
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
5. It was enjoyable to collaborate in this Biology coding workshop
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
6. The biological tasks were very interesting
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
7. It was easy to understand the instructions
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
8. It took a long time to learn to use the app
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
9. This app is difficult to use
   – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
10. The app is clear
    – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
11. This app is fun to use
    – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
12. The app easily does what I want
    – Strongly disagree - Disagree - Neutral - Agree - Strongly agree
13. I would like to use this app in school

  - Strongly disagree - Disagree - Neutral - Agree - Strongly agree
14. I would like to use this app outside of school
  - Strongly disagree - Disagree - Neutral - Agree - Strongly agree
15. The app has apparent faults
  - Strongly disagree - Disagree - Neutral - Agree - Strongly agree
16. If so, please explain why (open-ended question)

References

1. Andersen, R., Mørch, A.I., Litherland, K.T.: Learning domain knowledge using block-based programming: Design-based collaborative learning. In: Fogli, D., Tetteroo, D., Barricelli, B.R., Borsci, S., Markopoulos, P., Papadopoulos, G.A. (eds.) End-User Development. pp. 119–135. Springer International Publishing, Cham (2021)
2. BioCode: Learn bioinformatics, https://www.biocode.ltd/
3. Blum, L., Cortina, T.J.: Cs4hs: An outreach program for high school cs teachers. In: Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education. p. 19–23. SIGCSE '07, Association for Computing Machinery, New York, NY, USA (2007). https://doi.org/10.1145/1227310.1227320
4. Buckley, B.C.: Model-based learning. Encyclopedia of the sciences of learning **5**, 2300–2303 (2012)
5. Davis, F.D.: A technology acceptance model for empirically testing new end-user information systems: Theory and results. Doctoral dissertation, Massachusetts Institute of Technology (1985), https://dspace.mit.edu/handle/1721.1/15192
6. Eurostat: 70% of EU girls had basic or above basic digital skills, https://ec.europa.eu/eurostat/en/web/products-eurostat-news/w/edn-20230427-1
7. Eurostat: Eu had almost 7 million female scientists in 2021, https://ec.europa.eu/eurostat/web/products-eurostat-news/w/ddn-20230210-1
8. Gallagher, S.R., Coon, W., Donley, K., Scott, A., Goldberg, D.S.: A first attempt to bring computational biology into advanced high school biology classrooms. PLoS Computational Biology **7**, 1–7 (2011), 10.1371/journal.pcbi.1002244
9. Goldberg, D.S., Grunwald, D., Lewis, C., Feld, J.A., Hug, S.: Engaging computer science in traditional education: The ecsite project. In: Proceedings of the 17th ACM Annual Conference on Innovation and Technology in Computer Science Education. p. 351–356. ITiCSE '12, Association for Computing Machinery, New York, NY, USA (2012). https://doi.org/10.1145/2325296.2325377
10. Gupta, V., Irimia, J., Pau, I., Rodríguez-Patón, A.: Bioblocks: Programming protocols in biology made easier. ACS synthetic Biology **6**, 1230–1232 (2017), https://doi.org/10.1021/acssynbio.6b00304
11. Harrison, A.G., Treagust, D.F.: A typology of school science models. International journal of science education **22**(9), 1011–1026 (2000)
12. auf der Heide, F.M., Curzon, P., McOwan, P.W.: Computational Thinking; Die Welt des algorithmischen Denkens – in Spielen, Zaubertricks und Rätseln. Mathematische Semesterberichte 2019 66:2 **66**, 259–260 (2 2019). https://doi.org/10.1007/S00591-019-00249-0
13. Lockwood, J., Mooney, A.: Computational thinking in education: Where does it fit? a systematic literary review (2017)
14. Maier-Rabler, U.: Frauen als Nicht-(Mit-)Gestalterinnen der digitalen Transformation. Die digitale Transformation der Medien pp. 95–116 (2022)

15. Musante, K., DeWalt, B.: Participant Observation: A Guide for Fieldworkers. AltaMira Press (2010), https://books.google.at/books?id=ymJJUkR7s3UC
16. Sabitzer, B.: A neurodidactial approach to cross-curicual open learning. cool informatics. Habilitation (2013)
17. Schmidthaler, E., Stäter, R., Cápay, M., Ludwig, M., Lavicza, Z.: Easy coding in biology: Pilot workshop design and experiences from block-based programming with colette in secondary education. Journal of Research in Science, Mathematics and Technology Education **6(SI)**, 177–206 (2023). https://doi.org/https://doi.org/10.31756/jrsmte.619SI
18. Schwarz, C.V., Reiser, B.J., Davis, E.A., Kenyon, L., Achér, A., Fortus, D., Shwartz, Y., Hug, B., Krajcik, J.: Developing a learning progression for scientific modeling: Making scientific modeling accessible and meaningful for learners. Journal of Research in Science Teaching: The Official Journal of the National Association for Research in Science Teaching **46**(6), 632–654 (2009)
19. Statistik Austria: Bildung in Zahlen – Tabellenband, https://www.statistik.at/fileadmin/pages/325/Bildung_in_Zahlen_20_21_Tabellenband.pdf
20. Statistisches Bundesamt: Studienverlaufsstatistik 2022, https://www.destatis.de/
21. University, C.S.D.: Biology meets programming. bioinformatics for beginners, https://www.coursera.org/learn/bioinformatics
22. University of Vienna: Zahlen, Daten und Fakten zur Universität Wien. Gender im Fokus, https://personalwesen.univie.ac.at/organisationskultur-gleichstellung/publikationen/#c486668
23. Vasconcelos, L., Kim, C.: Preparing preservice teachers to use block-based coding in scientific modeling lessons. Instructional Science **48**, 765–797 (2020), https://doi.org/10.1007/s11251-020-09527-0
24. Vetter, T.R.: Descriptive statistics: Reporting the answers to the 5 basic questions of who, what, why, when, where, and a sixth, so what? Anesthesia & Analgesia **125**, 1797–1802 (2017)
25. Weintrop, D., Wilensky, U.: Comparing block-based and text-based programming in high school computer science classrooms. ACM Trans. Comput. Educ. **18**(1) (oct 2017). https://doi.org/10.1145/3089799, https://doi.org/10.1145/3089799
26. Wing, J.M.: Computational thinking. Communications of the ACM **49**, 33–35 (2006). https://doi.org/10.1145/1118178.1118215
27. Xu, Z., Ritzhaupt, A.D., Tian, F., Umapathy, K.: Block-based versus text-based programming environments on novice student learning outcomes: a meta-analysis study. Routledge **29**, 177–204 (7 2019). https://doi.org/10.1080/08993408.2019.1565233
28. Yamashita, S., Tsunoda, M., Yokogawa, T.: Visual programming language for model checkers based on google blockly. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **10611 LNCS**, 597–601 (2017)
29. Zurich University: Programming in biology, https://mnf.openedx.uzh.ch/courses/