

EDITORIAL

The elephant in the room—Should we be teaching coding to basic science students?

While it is difficult to determine what the public imagines a biochemist looks like, Artificial Intelligence (AI) is more than willing to try. Asked to return nine images of a biochemist, ChatGPT¹ returned eight images of people in white lab coats working in a lab with lots of plants or green liquids, five of which were looking through microscopes. As educators, teaching our biochemistry and molecular biology (BMB) students how to work with their hands in the laboratory is a priority. They learn to use pipettors to deliver consistent and accurate volumes of liquids. They learn to clone and purify DNA fragments, to express and purify proteins, and to perform kinetic studies. These are all important skills, but as is notably absent in the AI images, teaching labs often omit the level of computation that modern biochemistry requires. Seemingly every biological subfield now has an omics counterpart that examines not one piece of datum but thousands or millions of data points—genomics, proteomics, metabolomics, transcriptomics—and each of these fields relies heavily on software, algorithms and programming skills in languages such as Python, R, and C++.²

A 2023 report on “Building computational literacy through Science, Technology, Engineering, and Mathematics (STEM) education: A guide for federal agencies and stakeholders,”³ which primarily addresses K-12 education, makes several poignant recommendations including:

1. More initiatives for educators that help foster integration of computational literacy into existing teaching.
2. That agencies “advocate for tools and applications that demonstrate computational literacy components that are accessible to all.”
3. The promotion of computational literacy as a path to well-paying and interesting STEM careers.

Although teaching coding in basic science courses seems cutting edge, articles on this topic first appeared in the *Journal of Chemical Education* in the 1960's, and by the 1980's we find papers introducing students to coding in General Chemistry.⁴ McDonald notes that what was considered best practice in teaching coding in science courses then remains the same today, namely that coding

is an important tool for solving problems that cannot be solved using calculators or Excel.⁴

As educators, we seek to provide experiential learning that will benefit future generations of scientists. Given the power and ubiquity of BMB omics subfields, learning to handle big data (datasets too large to be handled without specialized programs or software) with some rudimentary coding skills is an important aspect of this training. Students matriculate to BMB programs with disparate experiences that influence their attitudes towards and learning of coding skills.⁵ Including coding-based or computationally focused labs as part of the BMB curriculum provides an option for students to inexpensively participate in cutting-edge, authentic research experiences that have been demonstrated to improve retention for underrepresented minorities in STEM fields.⁶ Recent development of readily available computational resources such as GitHub,⁷ Jupyter lab,⁸ R Studio,⁹ Google Colab,¹⁰ Chemcompute.org and MolSSI¹¹ enables faculty to deliver robust coding activities that are simple to follow and run in a classroom, yet can be easily modified to tackle new and interesting problems.¹²

Including coding in the BMB curriculum will serve to introduce students to BMB activities beyond white lab coats and microscope work, better preparing our students for big data BMB omics careers, and has the potential to increase diversity in the STEM workforce.

Despite its importance, we know firsthand the challenges integrating new technologies and content into a curriculum. As such, we are interested in your thoughts and experience regarding coding in your teaching and research, and ways you would like to grow. We plan to share the results of this survey in a future issue of BAMBED.

BMB coding survey link: <https://forms.gle/QFQjiPFnwwP58rHG9>





DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Walter Novak¹

Paul Craig²

Michael Foster²

¹Wabash College, Crawfordsville, Indiana, USA

²Rochester Institute of Technology

Correspondence

Walter Novak, Wabash College, 301 W. Wabash Ave,
Crawfordsville, IN 47933, USA.

Email: novakw@wabash.edu

ORCID

Walter Novak <https://orcid.org/0000-0002-1311-5953>

Michael Foster <https://orcid.org/0000-0001-6710-9156>

REFERENCES

1. ChatGPT (Version ChatGPT-4-turbo). Computer software. 2024 OpenAI.
2. Poinsignon T, Poulain P, Gallopin M, Lelandais G. Working with omics data: an interdisciplinary challenge at the crossroads of biology and computer science. In: Colloit O, editor. Machine learning for brain disorders. Volume 197. New York: Humana; 2023. p. 313–30.
3. Education FS. Building computational literacy through STEM Education. 2023.
4. McDonald AR. Teaching programming across the chemistry curriculum: a revolution or a revival? Teaching programming across the chemistry curriculum. Volume 1387. Washington, DC: American Chemical Society; 2021. p. 1–11. <https://doi.org/10.1021/bk-2021-1387.ch001>
5. Medeiros RP, Ramalho GL, Falcao TP. A systematic literature review on teaching and learning introductory programming in higher Education. IEEE Trans Educ. 2019;62(2):77–90. <https://doi.org/10.1109/TE.2018.2864133>
6. Estrada M, Burnett M, Campbell AG, Campbell PB, Denetclaw WF, Gutiérrez CG, et al. Improving underrepresented minority student persistence in STEM. CBE—Life Sci Educ. 2016;15(3):es5. <https://doi.org/10.1187/cbe.16-01-0038>
7. GitHub: Let's build from here. GitHub. 2024 <https://github.com/>
8. Kluyver T, Ragan-Kelley B, Pérez F, Granger B, Bussonnier M, Frederic J, et al. Jupyter notebooks—a publishing format for reproducible computational workflows. In: Loizides F, Schmidt B, editors. Positioning and power in academic publishing. Amsterdam: Players, Agents and Agendas; 2016. p. 87–90 <https://www.iospress.com/catalog/books/positioning-and-power-in-academic-publishing-players-agents-and-agendas>
9. RStudio Team. RStudio: integrated development environment for R. Boston: RStudio, PBC; 2020 <http://www.rstudio.com/>
10. Welcome To Colab—Colab. Welcome To Colab—Colab. 2024 <https://colab.research.google.com/>
11. MolSSI. The Molecular Sciences Software Institution. 2024 <https://molssi.org/>
12. Craig PA, Nash JA, Crawford TD. Python scripting for biochemistry and molecular biology in Jupyter notebooks. Biochem Mol Biol Educ. 2022;50(5):479–82. <https://doi.org/10.1002/bmb.21676>