18/1 (2020), 19–30
DOI: 10.5485/TMCS.2020.0470
tmcs@science.unideb.hu
http://tmcs.math.unideb.hu

Teaching
Mathematics and
Computer Science

Introductory Computer Programming Courses in Mathematics Curriculum

Leslie Jones and Tim Smith

Abstract. We present the results of surveys and curricular research on introductory computer programming courses that are required or recommended for mathematics degrees at U.S. colleges and universities. Our target schools were those with populations between $5,000$ and $20,000$ undergraduate students. A key result is a synopsis of programming languages in use in these introductory courses with Java, Python and $C++$ holding the top three spots. We found that 85% of the 340 schools in our pool require or recommend an introductory programming course as a component of a mathematics degree. Furthermore, most of these introductory programming courses are taught by faculty outside of the mathematics department. These results indicate that mathematics faculty value computer programming and should be actively involved in setting learning outcomes, incorporating skills and concepts learned in introductory programming courses into subsequent mathematics courses, and determining programming languages in use.

Key words and phrases: mathematics curriculum, introductory programming courses, programming languages.

MSC Subject Classification: 97D30, 97P20, 97P40.

Introduction

The objective of this research is to investigate the current state of computer programming in mathematics curriculum within U.S. colleges and universities. Guiding this research were four questions. Firstly, with what frequency is an introductory computer programming course required of mathematics students?

Secondly, are programming courses taught by mathematics faculty or faculty from other departments? Thirdly, what programming languages are in use? Lastly, how stable is the programming language choice and what philosophies determine the language chosen?

Surveys have been conducted and data gathered on introductory programming courses in other countries. Mason and Simon reported that Python and Java were the top programming languages in use in Australasia in 2016 with student numbers in these classes "on the rise" (Mason & Simon, 2017). Avouris studied computer programming courses in Greece in 2018 and found C to be the most popular language, with MATLAB in second place. Participation in programming classes had a steady trend over the six-year period from 2012 to 2018 (Avouris, 2018). In the United Kingdom, Murphy, Crick and Davenport published survey results in 2017, that pointed to Java as the main programming language in use in introductory programming courses although Python was perceived by the survey participants to be "easier to teach as well as to learn" (Murphy, Crick, & Davenport, 2017). In the United States, Richard Reid of Michigan State University started The Reid List, which documented introductory programming course languages in use for computer science and information systems majors at 400 universities across the country, beginning in the early 1990's. The most recent review was published in 2016 by Siegfried, Siegfried and Alexandro and showed Java at the top of the list, with Python and $C++$ within a percentage point of each other, but at less than half the popularity of Java. For those interested in a historical perspective on the transformation of introductory programming course languages we recommend Siegfried, Siegfried and Alexandro (Siegfried, Siegfried, & Alexandro, 2016) and Siegfried, Greco, Miceli and Siegfried (Siegfried, Greco, Miceli, & Siegfried, 2012). For information on programming languages in industry, the Institute of Electrical and Electronics Engineers (IEEE) journal IEEE Spectrum in collaboration with Nick Diakopoulous developed a programming language ranking profile that is available on the web and allows the user to customize the parameters (Diakopoulous, 2019). Additional programming language rankings and discussions can be found at the websites for RedMonk, 2019; PYPL Index, 2019; and TIOBE, 2019.

Our results are unique for their focus on mathematics curriculum and the perspective of mathematics faculty. We answer the principal questions above, provide direct faculty feedback and share insights gained through our research.

Methods

We defined our population as public and private colleges and universities with $5,000$ to $20,000$ undergraduate students who offer a four-year degree in mathematics. The selection of this population was motivated by the need to inform decisions on curriculum involving computer programming and the mathematics curriculum at our University, which has a population of approximately $9,000$ students.

To identify our population sample, we used the search engine of the National Center for Education Statistics (NCES) website to create a list of schools to survey (National Center for Education Statistics, 2019) Accessible faculty contact information was required. We developed a survey instrument and administered it to a randomly chosen test group of size twenty. Feedback from this survey was solicited as part of the participant responses. This feedback was then incorporated into the final survey. In total 340 colleges and universities were in the pool. Surveys were administered at the end of fall semester 2018 and the beginning of spring semester 2019. Surveys were initially sent to the chair of each department. Each recipient was asked to forward the survey in the case of a lack of familiarity with the introductory programming course. Ninety-five of the recipients actively participated in the survey (28%) with no obvious bias as to school size, location or public versus private.

For those schools in our pool that did not complete the survey, we searched information available through the internet to determine if a computer programming course appeared in the mathematics curriculum as required or recommended, the title of the introductory programming course as a potential identifier of the department in which it was housed and the language in use. When information was not found, we emailed faculty associated with the course per internet tags. The required category means a mathematics degree(s) offered by the university lists a computer programming course in the degree requirements for that major. We did not consider degrees that contained the words “computer science” such as a mathematics degree with a concentration in computer science. This is because we wanted to know if mathematics faculty valued computer programming as a part of mathematics curriculum as opposed to a degree where students chose the path knowing programming would be a significant part of the curriculum. We placed courses in the recommended category if a computer programming course was among a list of required electives. Other common disciplines in elective pools were physics and statistics. If an upper-level course was required or recommended, we researched the prerequisite introductory programming course as this was the focus of our study. We used degree guides and catalogs with great success as they are generally open access.

In determining the programming language in use for schools that we researched on the internet, we applied the following rules and search criteria: (1) a syllabus for the course from a semester beginning no earlier than fall 2017 which referenced the programming language, (2) a book for the course with the name of the programming language in the title on reserve at the bookstore, which was information only available for spring 2019 and fall 2019, (3) a course description in the school catalog which explicitly named the language, and (4) personal email correspondence with a faculty member associated with the course. We summarize our results from the surveys as well as the information we gleaned from the internet and personal correspondence. Total response counts from the surveys for individual questions are provided.

Results

Of the 340 schools in our study, 289 (85%) require or recommend a computer programming course for one of the degrees in their mathematics portfolio excluding those which are mathematics/computer science degrees (Figure 1). This is a strong indicator that programming skills are valued by mathematics faculty.

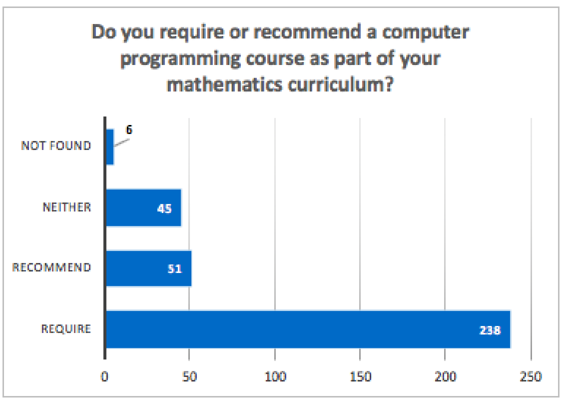

Figure 1. Answers to the question of whether a programming course is required or recommend. Includes results for both survey respondents and online researched schools.

While programming courses overwhelmingly appear in mathematics curriculum, more of these courses are taught by faculty from other departments. The surveys garnered 92 answers and showed that 26% of the courses were taught by faculty in the mathematics department only, 57% were taught by faculty exclusively in another department and 17% were taught both by faculty in the mathematics department and faculty in another department (Figure 2). For the schools researched online, we were only able to determine the prefixes of the programming courses and not the departmental home of the faculty of the 165 schools for which we found the programming language. The breakdown was 87% with CS or its derivative in the course prefix or prefixes as the only course(s) allowed (e.g. students only had the option to take a course with a CS prefix to satisfy this requirement or recommendation), 4% had a math prefix in the course prefix or prefixes as the only course(s) allowed, 1% had a combined math/computer science prefix or combined engineering/computer science prefix, 4% had a list of choices that included either math and CS or CS and other, and 3% gave only options of courses with prefixes in the other category. The other category consisted of ITEC(2), ELEG(1), numerical identifier only(1), ENGR(2), EGR(1) and IT(1). It may be that mathematics faculty of the researched schools are teaching the CS-prefixed courses with the same frequency as the survey results. However, we suspect that the population of faculty who completed the survey had a higher representation among teachers of the introductory programming courses than non-participants.

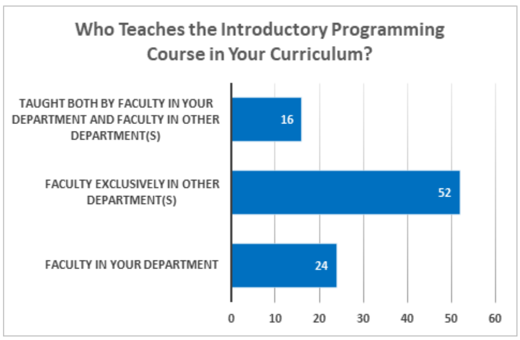

Figure 2. Summary of responses to the survey question about who teaches the introductory programming course listed in the curriculum.

The programming language results show three prominent languages: Java, Python and $C++$. The fourth most frequent category is choices and the fourth language, following at a distance, is MATLAB (©1994 The Math Works, Inc.).

The choices category represents cases where more than one course is given as an option to fulfill the requirement and different languages are taught in those courses or the programming language is said to vary by faculty or semester giving students de facto choices. In the choices category, the most frequently listed languages are also Java, Python and $C{+}{+}$. Another category with more than one language listed is called multiple and counts single courses that include multiple languages. For the results from our online search, Java was the leader with 40 appearances followed by Python with 38 and $C++$ with 29. From the survey results, Java leads the pack with 31 appearances, $C++$ is in second place with 17 and Python is close behind at 16. Figure 3 is a chart of the combined online researched and survey occurrences of each language excluding the choices and multiple categories and Figures 4 and 5 are charts for the choices and multiple conglomerates respectively. A table with a breakdown of languages by category is given in the appendix.

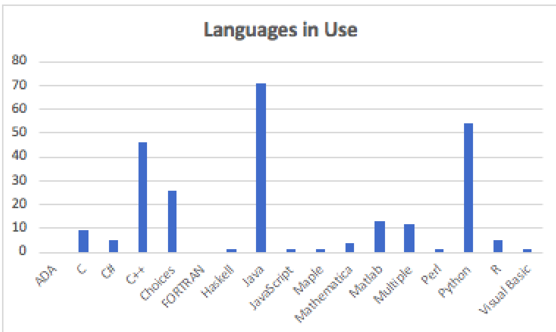

Figure 3. Languages in use in introductory programming courses for the 238 schools for which this information was found excluding the *choices* and *multiple* categories.

Below are sentiments from mathematics faculty about the choice of programming language in use in courses required and recommended for mathematics majors. These comments were expressed to us via email and survey responses and highlight the diversity of thinking on this topic among mathematics faculty.

"For mathematics students, I certainly would consider NOT using Java as an introductory course. A course in Python/C would be much more appropriate" (private email).

"[Python is] better suited to a mathematics major with a concentration in data science" (survey comment).

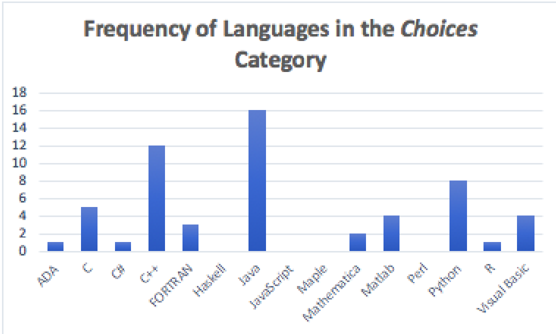

Figure 4. Breakdown of the *choices* category.

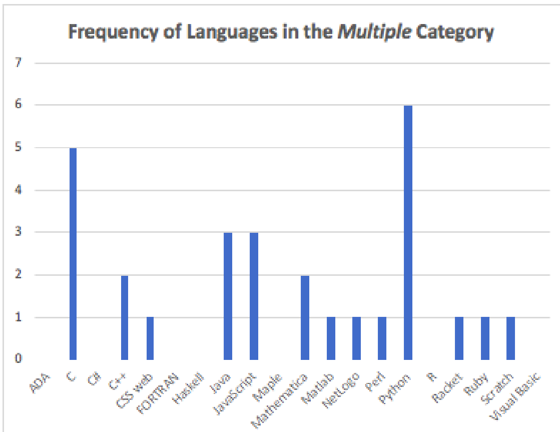

Figure 5. Breakdown of the *multiple* category.

"... we also think that both R and Python are better job credentials than Java" (survey comment).

"As a mathematics department, we are more interested in the process of learning to write code than the language itself" (survey comment).

The following results are from survey responses only because the information was not available online. The responses to the question as to whether the department planned to change the current programming language in the introductory course were 88% "no" and 12% "yes" for $N = 67$. This indicates strong stability for the language in use. The results of the question about introductory programming class size, with $N = 70$, yielded the following: 10% reported a class size of $10 - 19$ students, 34% reported $20 - 29$ students, 44% reported $30 - 39$ students, 7% reported $40 - 49$ students, 1% (1 response) listed 150 students and 3% (2 responses) listed 500 students. The final question for which we give results is "How would you characterize student demand for this course?" With $N = 68$, the responses were 44% "increasing," 3% "decreasing", and 53% "neither increasing nor decreasing."

Discussion

Our results show that college mathematics students are taking introductory programming courses with 85% of the 340 schools in the cohort requiring or recommending a programming course in the curriculum. Further, the demand for these courses is perceived to be stable or increasing. Most schools in our study outsource the teaching of introductory programming courses. The top three programming languages in use in the introductory programming courses listed as required or recommended are Java, Python and $C++$. This follows industry trends with IEEE Spectrum, RedMonk, PYPL and TIOBE listing Java, Python and $C++$ in the top six languages in use in industry (Diakopoulous, 2019; RedMonk, 2019; PYPL, 2019; TIOBE, 2019).

There is no indication from our results that these languages are being considered for replacement at other than a handful of universities. The AP Computer Science A exam administered by the College Board tests students on Java fundamentals (College Board, 2019). The continued use of Java in the AP Computer Science course in high school may impact the decision of college programs to teach Java in introductory programming courses.

The sentiments from mathematics faculty expressed in email and open-ended survey responses raise an interesting question that Mathematics departments must face. That is, what should be the drivers of choosing a particular programming language. What makes one language potentially "better" than another when applied to a mathematics curriculum? Demand within industry is an obvious consideration. However, while many sources point to frequency of use, this demand is not evenly distributed, with some languages being favored for certain tasks such as C++ for game development or Python for data science. On the other hand, if we see programming as an avenue to deepen mathematical understanding and problem-solving skills, then a language that best supports algorithm design and computational thinking should be chosen. What produces better thinking in our students may not align with current industry trends which focus heavily on momentum, efficiency and productivity. Mathematics faculty must examine whether the language in use in introductory programming courses they require demonstrates appropriate constructs, applications and ideas for future work by mathematics students.

Questions also remain as to how the programming skills and concepts learned by mathematics students are incorporated into upper-level mathematics courses, with what frequency, and by whom. Cline, Fasteen, Francis, Sullivan and Wendt share their integration plan for programming in the mathematics curriculum in their article: "Integrating Programming Across the Undergraduate Mathematics Curriculum" (Cline, Fasteen, Francis, Sullivan, & Wendt, 2019). Jones and Hopkins provide a template for a course that teaches introductory programming in a mathematics course with a focus on mathematical algorithms (Jones & Hopkins, 2019).

Conclusion

This research establishes the prevalence of introductory programming courses in college mathematics curriculum across the United States. Further, most introductory programming courses are taught by faculty outside of the mathematics department. The data serve as a potential call for more involvement by mathematics faculty in decisions related to introductory computer programming courses to optimize the learning potential.

References

Avouris, N. (2018). Introduction to computing. In *Proceedings of the 22nd pan-hellenic conference on informatics - PCI 18.* ACM Press. Retrieved from `https://doi.org/10.1145/3291533.3291549` doi: 10.1145/3291533.3291549

Cline, K., Fasteen, J., Francis, A., Sullivan, E., & Wendt, T. (2019, August). Integrating programming across the undergraduate mathematics curriculum. *PRIMUS*, 1–15. Retrieved from `https://doi.org/10.1080/10511970.2019.1616637` doi: 10.1080/10511970.2019.1616637

College Board. (2019, September). *Ap computer science a.* Retrieved from `https://apstudents.collegeboard.org/courses/ap-computer-science-a`

Diakopoulous, N. (2019, July). *Interactive: The top programming languages 2018. ieee spectrum.* Retrieved from https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2018

Jones, L. B., & Hopkins, B. J. (2019, July). Teaching a course in mathematical programming. *PRIMUS*, *30*(5), 571–600. Retrieved from https://doi.org/10.1080/10511970.2019.1619207 doi: 10.1080/10511970.2019.1619207

Mason, R., & Simon. (2017). Introductory programming courses in australasia in 2016. In *Proceedings of the nineteenth australasian computing education conference on - ACE17.* ACM Press. Retrieved from https://doi.org/10.1145/3013499.3013512 doi: 10.1145/3013499.3013512

Murphy, E., Crick, T., & Davenport, J. H. (2017, April). An analysis of introductory programming courses at UK universities. *The Art, Science, and Engineering of Programming*, *1*(2). Retrieved from https://doi.org/10.22152/programming-journal.org/2017/1/18 doi: 10.22152/programming-journal.org/2017/1/18

National Center for Education Statistics. (2019). *College navigator.* Retrieved from https://nces.ed.gov/collegenavigator/

PYPL. (2019, July). *Pypl popularity of programming language. pypl index.* Retrieved from http://pypl.github.io/PYPL.html

RedMonk. (2019, July). *The redmonk programming language rankings: January 2019.* Retrieved from https://redmonk.com/sogrady/2019/03/20/language-rankings-1-19/

Siegfried, R., Greco, D., Miceli, N., & Siegfried, J. (2012). Whatever happened to richard reid's list of first programming languages? *Information Systems Education Journal*, *10*(4), 24–30.

Siegfried, R., Siegfried, J., & Alexandro, G. (2016). A longitudinal analysis of the reid list of first programming languages. *Information Systems Education Journal*, *14*(6), 47–53.

TIOBE. (2019, July). *Tiobe index for july 2019.* Retrieved from https://www.tiobe.com/tiobe-index/

APPENDIX

	Researched Online	Survey Results
ADA	0	0
C	8	1
C#	5	0
C++	29	17
Choices	21	5
FORTRAN	0	0
Haskell	1	0
Java	40	31
JavaScript	1	0
Maple	0	1
Mathematica	1	3
MATLAB	4	9
Multiple	12	0
Perl	1	0
Python	38	16
R	3	2
Visual Basic	1	0

Table 1. Frequency of languages researched and from surveys

	Researched Online	Survey Results
ADA	1	0
C	4	1
C#	1	0
C++	10	2
FORTRAN	3	0
Haskell	0	0
Java	11	5
JavaScript	0	0
Maple	0	0
Mathematica	1	1
MATLAB	4	0
Perl	0	0
Python	6	2
R	0	1
Visual Basic	4	0

Table 2. Frequency of languages in *choices* category

	Researched Online	Survey Results
ADA		
C	4	1
C#		
C++	2	
CSS for web development	1	
FORTRAN		
Haskell		
Java	3	
Maple		
Mathematica		
MATLAB	1	
NetLogo	1	
Perl	1	
Python	5	1
R		
Racket	1	
Ruby	1	
Scratch	1	
Visual Basic		

Table 3. Frequency of languages researched and from surveys

Leslie Jones
The University of Tampa, Department of Mathematics
401 W.Kennedy Blvd., Tampa, Fl 33606

E-mail: `lbjones@ut.edu`

Tim Smith
The University of Tampa, Infomartion Technology management
401 W.Kennedy Blvd., Tampa, Fl 33606

E-mail: `tcsmith@ut.edu`

(Received December, 2019)