Development of a Parallel Stretch Programming Course to Improve Outcomes for Students with Minimal Computing Experience

Nicholas Baine, Ph.D., P.E.
School of Engineering
Grand Valley State University
Grand Rapids, MI 49504
Email: bainen@gvsu.edu

Introduction

This is a work-in-progress paper that introduces a new initiative to improve student success in an introductory programming course for engineering students. Engineering students, regardless of major, need to learn how to program. Most engineers will never write a program that the general public will interact with, but they will need to run simulations and develop new computational tools and models to do analysis and make data driven design decisions. Interacting with advanced computing tools requires a high level of computer literacy. K-12 educators are working to provide additional enrichment in this area, but it tends to be elective and provided if resources allow.

Given that schools are providing different opportunities if any, students entering college-level engineering and computing programs are arriving with a growing diversity of computing experience. This leads to a frustratingly high range of student capability for introductory engineering programming courses, making it difficult to design the course to best serve all students in the course.

There are a couple options to help resolve this issue. The first is to create a new preparatory course to provide those with little prior computing experience. The second is to take the existing first computing course and offer a stretch version (2 semesters instead of one). We chose to develop the latter option. To create a stretch version, the content of the original course is spread out over two semesters (two courses) with more examples and class time devoted to the content. The slower delivery allows students more time to ask questions, get assistance, and complete programming assignments.

This paper describes the structure of this parallel stretch course sequence and the subsequent addition of a parachute option, enabling students who struggle early in the one-semester course to switch to the two-semester stretch course. We are also allowing students who struggled in the second half of the one-semester course to take the second stretch course the following semester instead of repeating the full course. Rather than repeat all the content, they can spend an entire semester mastering the topics they struggled with.

The paper concludes with thoughts about the benefits of this approach, which is primarily increased flexibility enabling students with diverse backgrounds to have a path that best suits their needs. The main challenges come with advising students and identifying students who would benefit from the new stretch option.

The Problem

There is a myth that students who grow up in the internet age will be better equipped and savvy. Studies now warn us about the negative consequences in assuming students have digital skills simply because they have grown up with computers [1]. The OECD (Organization for Economic Co-operation and Development) conducted global research in 33 developed countries and reported that only 5% of the general population possesses high computer-related skills and only 30% can address medium-complexity tasks [2].

Depending on the secondary school, there may be classes offered on a variety of topics, which include office productivity applications, website design, and traditional programming. In the best of scenarios, the school has a robust path to learn a programming language such as Python, Java, or C++. They may also offer Advance Placement (AP) Computer Science A or AP Computer Science Principles. The former is a traditional object-oriented programming course and is taught in Java, and the latter is less about programming and focusses on the application of computing in the world [1]. While AP courses provide a clear indication of whether course credit is warranted at the college level, the other opportunities can provide a valuable foundation for learning computing topics in college. Unfortunately, these opportunities are not consistently offered nor are they universally available. The result of these differing opportunities is a growing diversity of preparation among incoming college students, including those seeking to attend a STEM program.

At Grand Valley State University, we are seeing the result of this growing diversity in our classrooms, and it is a driving force in students leaving engineering school. Students leave STEM majors for a variety of reasons, but the most common contributing factors are low achievement, alienation, boredom, and a lack of engagement [4, 5]. A White House report also cited experiences with uninviting atmospheres, participating in weed-out courses, and taking courses with no obvious relevance as reasons for leaving a STEM major [6]. This leads to the obvious question: How can a course be engaging and not boring for students with a strong general computing background yet not make students with a weak background perceive a course to be weeding them out as they grapple with feeling unwelcome and low achieving? The answer has yet to come, but instructors have done their best as courses have been adjusted and modified with optional resources made available to those who need them. The problem is that a single course will always be a compromise.

Two Course Paths

At Grand Valley State University, all engineering students are required to learn how to program, which was done in a single semester course in the first year with a co-requisite of Calculus I; however, a significant portion of the students struggle with non-programming tasks such as navigating computer directories. They also take extra time to understand programming tools and are not accustomed to software that is not written for the general public.

One possible solution is to provide a prerequisite preparatory course on general computing. Such courses are available at most universities for non-STEM majors; however, most are focused on productivity applications such as MS Office. It is believed that a better approach is to slow down the delivery of the programming course. The result is a two semester “stretch” version of the original programming course. In these new courses, students are expected to complete the same reading and programming assignments; however, it is split over two semesters with additional time for instructor led examples, class activities, and guidance on your work. Table 1 shows a comparison of the two paths.

Table 1: Comparison of the Two Paths

Path One	***Path Two (Stretch)***
• One Semester • 2 Credits Total • Grade of C or Better fulfills requirement in Engineering Foundations • Assumes strong math skills • Requires a familiarity of computers	• Two Semesters • 4 Credits Total • Grade of C or Better in 2nd course fulfills requirement in Engineering Foundations • More time to practice writing programs and algorithm development • Additional support from instructors

The stretch version offers twice as much time for students to master topics before moving on. This is key for student success in programming as each new topic builds upon the topics that came before. The first course of the stretch sequence, introduces programming to the students, covering basic I/O, operators, expressions, logic expressions, conditional statements, and repetition statements. The second course builds upon that with topics on functions, file I/O, pointers, arrays, and sorting algorithms. As the students progress, they are required to write increasingly complex programs often with an engineering application.

The second path is being piloted in the 2021-2022 school year and is showing promise, but it has become apparent that not all the students who need the slower paced courses correctly chose that path. Many students were overly confident in their abilities and mistakenly enrolled in the faster paced Path One.

The Parachute Option

Without placement testing, there is no guarantee that all students will be placed in the best courses to ensure their success. Given the burden of testing, it was decided to allow students to self-assess and choose between Path One and Path Two; however, there are many who choose Path One who soon discover that they are falling behind. This discovery unfortunately occurs after traditional Add/Drop deadlines. To accommodate for this, a new policy was made that allowed for students to “parachute” from the course overwhelming them, into the stretch course.

This option is made available on week 3, with the deadline of the university’s 75% refund deadline. Students who are interested in the option must discuss it with their instructor and fill out a form requesting the option. The students then have their registration swapped with an open seat in the slower paced course if there are seats available on a first-come first served basis. Many sections are taught at the same time, making swaps logistically easy for most.

Not Repeating Path One

There are some students who were borderline successful (did not earn an F) taking Path One, but they failed to pass the course with a C or better as required for their major. This is most often the result of their struggle with the growing complexity in the second half of the course. Typically, a student would be required to repeat the course, but the stretch version provides an alternative. Students could choose to either repeat the Path One course, or they could take the second course of Path Two, which gives them an entire semester to focus on the topics they struggled with. This helps with engagement as they are not bored with introductory content they did not need to repeat, and it allows for more time on the topics they do need.

Discussion

At Grand Valley State University, there is a growing diversity in student preparation for an introductory programming course. Consequently, there is a clear need for more than one option for students to learn programming and develop their computing skills. This led to the development of two paths, an accelerated course offered in a single semester and a stretch version with the same topics offered over two semesters. This has increased flexibility for students, providing students who are ready for a fact paced course the engaging and challenging experience they want while not trapping and overwhelming students who are not ready for it.

This 2021-2022 school year is the first to offer the stretch version of the course along with the parachute option. Anecdotally, this new path is a success and is leading to improved student outcomes among students who are likely to fail in Path One and change their major.

Works Cited

[1] P. A. Kirschner and P. De Bruyckere, "The myths of the digital native and the multitasker," *Teaching and Teacher Education,* vol. 67, pp. 135-142, 2017.

[2] OECD, "Skills for a Digital World," OECD, Paris, France, 2017.

[3] CollegeBoard, "AP Computer Science," 2022. [Online]. Available: https://apstudents.collegeboard.org/courses/ap-computer-science-a. [Accessed January 2022].

[4] J. A. Fredricks, P. C. Blumenfeld and A. H. Paris, "School Engagement: Potential of the Concept, State of the Evidence," *Review of Educational Research,* vol. 74, no. 1, pp. 59-109, 2004.

[5] R. J. Swap and J. A. Walter, "An Approach to Engaging Students in a Large-Enrollment, Introductory STEM College Course," *Journal of the Scholarship of Teaching and Learning,* vol. 15, no. 5, pp. 1-21, 2015.

[6] President's Council of Advisors on Science and Technology (PCAST), "REPORT TO THE PRESIDENT -- PREPARE AND INSPIRE: K-12 EDUCATION IN SCIENCE, TECHNOLOGY, ENGINEERING, AND MATH (STEM) FOR AMERICA'S FUTURE," The White House, Washington, DC, 2010.