

Experience Report: Designing Massive Open Online Computer Science Courses for Inclusion

Sophia Krause-Levy, Mia Minnes, Christine Alvarado, and Leo Porter

University of California, San Diego

{skrausel,minnes,cjalvarado,leporter}@eng.ucsd.edu

ABSTRACT

Although Massive Open Online Courses have the potential to reach a much broader audience and offer a lower cost education than traditional in-person classes, they have struggled with low completion rates and low diversity amongst those enrolled and completing the courses. In 2015, we built a series of online courses in computing with the specific goal of attracting and retaining students from groups underrepresented in computing. In our design, we incorporated a number of features aimed at improving the inclusive nature of the courses including: a project-centered course design; an online version of Peer Instruction ConceptTests; videos where students, faculty, and professionals report their struggles when they first learned computing concepts; videos by professional software engineers explaining how computing concepts from the course are used in industry; and videos aimed at providing additional support on the project to students who might be struggling. In this work, we report on the design of the courses and examine how successful our courses were at attracting and retaining women students. We find that compared to other computing courses offered by our institution on the same platform, our courses have: a higher percentage of women enrollment, higher rates of course completion for both men and women, and a slightly smaller gap between completion rates for men and women.

CCS CONCEPTS

• **Social and professional topics** → **Computing Education.**

KEYWORDS

diversity, underrepresentation, gender, MOOCs, online computing education

ACM Reference Format:

Sophia Krause-Levy, Mia Minnes, Christine Alvarado, and Leo Porter. 2021. Experience Report: Designing Massive Open Online Computer Science Courses for Inclusion. In *26th ACM Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2021)*, June 26–July 1, 2021, Virtual Event, Germany. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3430665.3456340>

1 INTRODUCTION

Much has been written about the potential of Massive Open Online Courses (MOOCs) to revolutionize education in terms of opening

access to a broader range of students. While to some degree MOOCs have lived up to these expectations—indeed MOOCs have the ability to reach students around the globe [2, 3, 7, 11]—in many ways MOOCs are still limited in practice. Particularly in MOOCs in highly technical fields like computing, most learners are young adult men who already possess a bachelor’s degree or higher [2, 3, 7, 11, 13].

In 2015, we developed a series of intermediate programming MOOCs with the explicit goal of creating courses that would broaden participation from groups traditionally underrepresented in computing in general, and MOOCs in particular, including women, and Black and Latinx learners. To address this goal, we integrated best practices from in-person learning into the online space. These best practices included: contextualized and project-based courses, modified ConceptTests from Peer Instruction (PI), normalizing struggle, and support videos.

This paper describes the explicit design of our courses around these best practices and how we adapted each practice to the online MOOC space. We analyze student behaviors and outcomes in the courses for all students and by gender. We had hoped also to analyze participation and outcomes by race, but unfortunately we did not have enough data to yield meaningful insights. Reporting both gender and race was optional, and the Coursera platform attempted to infer gender but not race when it was not provided. Our analysis examines course enrollments, course completion rates, and which components of the courses were most commonly engaged with by learners. For course enrollments and completion rates, we compared our courses against other computer science (CS) MOOCs offered by our institution on Coursera. Encouragingly, relative to these other CS MOOCs, we found higher overall completion rates, higher completion rates for men and women, and higher proportions of women among the paid enrollments in our courses. Despite this encouraging result, the proportion of women in our MOOCs did not exceed the (rather low) rates of women majoring in computer science in our country [20]. We hope the lessons learned described in this experience report can be a stepping stone for further innovation.

2 BACKGROUND

2.1 Massive Open Online Courses (MOOCs)

Learners enroll in MOOCs for a number of reasons, with some of the top reasons being: for enjoyment or challenge, because they find the material in a course interesting, or to gain a certificate at the end of the course [5, 10, 11]. Due to MOOCs generally being self-paced and allowing students to jump between material, many students do not take the expected linear path through the course (for example, they do not complete content in the order it is provided) [7, 15].

A literature review on MOOCs by Hew and Cheung found that the common differences between MOOCs and traditional in-person



This work is licensed under a Creative Commons Attribution-ShareAlike International 4.0 License.

ITiCSE 2021, June 26–July 1, 2021, Virtual Event, Germany.

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8214-4/21/06.

<https://doi.org/10.1145/3430665.3456340>

learning are that MOOCs have much lower rates of course completion, lower instructor support and presence, and they tend to have a much larger and more international enrollment [10]. In light of these differences, DeBoer et al. suggest that MOOC creators may need to redefine measures of success for their courses and readjust their expectations to take into account the different backgrounds of students and students' intentions coming into a course [5].

Further illustrating this need, a study conducted by Littlejohn et al. examined the differences in motivations between students with high and low self-regulated learning in an "Introduction to Data Science" MOOC. They found that students with higher self-regulated learning were more likely to be taking the course to focus on mastery of the material and evaluated their learning based on its connection to their future needs in their profession. Students with low self-regulated learning focus more heavily on receiving a certificate at the end of the course and completion of all of the available material [15]. Similarly Guo and Reinecke [7] studied 140,546 students in four EdX MOOCs to better understand student skills and behaviors and found that the students who earned the course certificate skipped 22% of the course content on average and often did not follow the linear set up of the course, jumping between content. Students seem to develop their own structure for navigating through a course. The results from these studies show that there are different motivations and needs of students as they take a course. In designing our courses, we explicitly built in multiple paths for learners through the course. In this report, we find that learners in our courses indeed took advantage of these different paths.

2.2 Gender Representation in MOOCs

There is a known gap in gender enrollment in MOOCs, with women enrolling and completing courses at a lower rate than men [2, 3, 7, 11, 13]. A study by Breslow et al. found that in a "Circuits and Electronics" edX course 88% of the students who had completed the course and received the certificate were men, with 37% of the students having already attained their bachelors and 28% having already obtained their masters [3]. Kizilcec et al. similarly analyzed three computer science MOOCs and found women to have much lower enrollment and completion rates [11]. This suggests that while MOOCs have the ability to reach a much larger demographic, they mainly help students who have already spent time in higher education and are much more likely to attract men.

3 COURSE DESIGN

3.1 Course Descriptions

At the time the courses were designed, they were the second half of a larger project to create a series of courses that taught software engineering fundamentals on Coursera. The first half of the series was created by another institution and its focus was introductory CS concepts in Java, including typical CS0 and CS1 content and some initial discussion of data structures. Our courses were the second half in the series with the scope of teaching object-oriented principles, data structures (basic and advanced), basic algorithms, runtime analysis and testing, and how to prepare and succeed in a software engineering interview. A brief summary of these courses can be found in Table 1.

3.2 Course Design for Inclusion

Our overarching goal with the design of these courses was to integrate best-practices for inclusive learning from in-person classes into an online setting. These practices included: contextualized computing [8] and Project-Based Learning [6, 12, 19], normalization of struggle [18, 23], Peer Instruction [4, 17], and emphasizing the real-world value of the course concepts [14].

Project Based Learning in a Real-World Context. To help students understand the value of the CS concepts in the courses, as well as to empower them to see how these concepts could be used to build something real, each course was anchored by a project with real-world application. The application for each course is given in Table 1. Each new computing concept was motivated by the need to add functionality to the application. For example, in one course the project was to build an earthquake visualization tool that would interactively plot data about recent earthquakes on a world map. Inheritance and polymorphism were introduced based on the need to define basic functionality for a graphical "marker" that would display the earthquake data, and then to customize what is displayed depending on whether the earthquake was over the ocean or land. These real-world contexts were also adopted as a form of computing in context, based on the success of Media Computation in improving outcomes for women [9].

Recently, Lewis et al. showed that women and first-generation students are more likely to have goals to improve society than other students and that perceived goal alignment with the field of computing impacts students' decisions to remain in the field [14]. In addition to having the course revolve around a meaningful project, we added a series of videos called "In the Real World" where professional software engineers described how they used concepts from the class in their jobs building software tools used by a large population of users.

Normalizing Struggle. We were inspired by work from the University of Texas where their intervention to aid retention of students from underrepresented groups in STEM had included students reading essays from upper-classmates talking about challenges they overcame early in their college careers [23]. To incorporate the idea that everyone struggles at some point into our course, we created a video series called "When I Struggled" which featured computer scientists talking about a point when they struggled to learn concepts from the course or in school in general. In addition to videos of ourselves talking about our struggles, we included videos from students at our institution and professional software engineers.

Peer Instruction. Peer Instruction had been shown to be a best practice for improving student outcomes in computing [16, 22, 25] and has since been shown to also improve outcomes for students from underrepresented groups in computing [21]. Peer Instruction is the process of students answering a question, discussing that question with peers, answering the question again, and the instructor leading a class-wide discussion about the solution. The direct interactions were not possible on Coursera at the time, particularly for a course offered asynchronously. However, we used students at our institution to aid with that component. For the peer discussion portion of the format, we filmed 3–4 students from our institution acting as though they were students in a class working through the problem. Students took turns such that each student offered a

Table 1: Online computer science courses described in this report.

#	Acronym	Title	Course Content	Project
1	OOP	Object Oriented Programming in Java	Memory Diagrams, GUIs, Inheritance and Polymorphism, Event Handling, Searching and Sorting	Real-Time Global Earthquake Visualization
2	DS	Data Structures and Performance	Runtime Analysis, Strings, Interfaces, Testing, LinkedLists, ArrayLists, Trees, Binary Search Trees, Tries, Hash Maps	Text Editor with Spell Check and Autocomplete
3	ADS	Advanced Data Structures in Java	Graphs, Search Algorithms (BFS, Dijkstra, A*), NP-Hard and NP-Complete Problems	Finding Directions on a Map w/Real Street Data
4	SE-Interview	Mastering the Software Engineering Interview	Technical Interview Expectations, Phone Interviews, Presenting Technical Accomplishments, Algorithmic Thinking	Self-Assessed Videos of Practice Interviews
5	ASND	Analyzing Social Network Data	Heuristics for Addressing NP-Hard Problems, Gathering Data from Social Networks	Capstone Project (with project suggestions)

Table 2: Video series in the course.

Title	Motivation
Project	PBL and Computing in Context
In the Real World (IRW)	Computing in Context
Concept Challenges (CC)	Peer Instruction
When I Struggled (WIS)	Normalizing Struggle
Support	Additional Project Help
Content	Standard Course Concepts

common misconception in one video but also got to be the student with the correct answer in another.

We faced several challenges in creating videos that would simulate a peer-instruction experience. The first challenge was that the videos almost always ended with a consensus toward the correct answer, which can often not be the case in an actual classroom. We made this choice because we felt it would be confusing to end these videos with the wrong answer, particularly for asynchronous learners who might not watch the follow up video to learn that the consensus answer was in fact wrong! A second challenge was that the students' discussion often became hard to follow from the outside. To help the online learners follow the discussion, we added code snippets and annotations in post processing to highlight the code being discussed at each point in the video by the students.

Using these videos from students, we put together a series of videos named "Concept Challenges" (based on ConceptTests from Peer Instruction [4]). These videos included: 1) the instructor posing a question in a video, 2) the online learner answering the question using an in-video quiz feature, 3) the video of "students" discussing the question, 4) the online learner answering the question again, and 5) the instructor walking through the solution to the problem. **Additional Support.** We were aware that not all students entered the online courses with equal backgrounds and that prior experience is a predictor of student success in early computing courses [24]. As such, we added additional content designed to help students who needed additional support in completing the projects and to offer another perspective on some of the courses' core concepts. These videos were called "Support" videos and we instructed students to watch these videos if they felt they wanted more help with the course content.

A summary of the types of videos that appeared in the courses is in Table 2. Each video was labelled with its type on Coursera; learners were encouraged to customize their path through the courses by engaging with the video types that best served them at each stage. In addition to the core course elements, we made a conscious effort

to ensure the majority of the course instructors were women and that all imagery of computer scientists for the class, including the videos from professional software engineers, represented a diverse population in terms of race and gender.

4 METHODS

4.1 Population

Our goal was to understand how students used the various components of our courses and to determine which students completed our courses, as compared to other CS MOOCs that did not incorporate the specific inclusion practices mentioned above. Of course, MOOCs are used for a wide variety of reasons by students [5, 10, 11], including just looking for a clear explanation of single concept. In our current analysis we elected to focus only on students that we believed had the intention of completing each course. We used paying for the course, either out of pocket or via financial aid, as a proxy for a student's intention to complete the course. We refer to this group as "paid-learners." As a baseline for comparison, we selected other CS MOOCs developed by our university. Thus the two populations in our study are: (1) paid learners who enrolled in other computer science (CS) MOOC courses created by our university and (2) paid learners who enrolled in the first four courses in our computer science course series designed for inclusion (CS-DFI).

The CS-DFI courses included in the second population include the first four courses listed in Table 1. We chose to exclude the final course in our series, Analyzing Social Network Data for two reasons: (1) low enrollments due to enrollment restrictions for courses considered Capstone courses (students must earn certificates for all prior courses to be eligible to enroll) and (2) the fact that the course is primarily self-driven by students without a large amount of instructor-created content.

In cooperation with Coursera and in accordance with our Human Subjects approved protocol, we were able to gain access to the enrollment and completion rates of paid learners and the gender breakdown of these students. In the case of the students in the CS-DFI courses we were able to access more detailed information about their participation in the courses.

4.2 Data Collection

The data we collected came from two sources, summarized in Table 3. The first dataset included completion rates for men and women for all computing courses (including ours) offered on Coursera by our institution and was provided by the online hosting company in May 2019. This data source is used to compare completion

Table 3: Datasets used in the evaluation.

#	Source	Dates	Description
1	Coursera	Inception–May 2019	completion rates for women and men among paid learners for all CS and CS-DFI courses offered on Coursera by our institution
2	Instructor Access	Inception–Dec 2019	Background demographics of learners (nationality, gender), assessment results, course content interaction (videos watched), course completion rates.

rates for our courses relative to other CS courses on Coursera. The second dataset was collected by the course instructors and includes data for our CS-DFI courses dating from inception to September 2020. This dataset contains considerably more details including student demographics, and interactions with the platform (e.g., videos watched). We use the second dataset for exploring student behaviors in our courses. For this dataset, we chose to exclude students who enrolled in 2020 to remove any changes in student behavior that may have occurred during the COVID-19 pandemic that first impacted the United States in February 2020.

When analyzing demographics, we chose to focus solely on gender as all demographic information was optional for students to complete and the number of students who identified from underrepresented racial and ethnic groups was too small to yield any meaningful insights. A student’s gender is determined either through the student reporting their gender via a survey on the online platform or through the platform inferring a student’s gender by their first name. The demographic dashboards at inception of the course only recorded two genders, providing the options “male” and “female”. An “other” category has been added. Since we are looking at historical data, we report only on those who selected “male” or “female,” and throughout this paper we use the more correct terms “men” and “woman” to refer to gender, assuming that students chose the option that most closely corresponded to their gender identity, not their biological sex.

5 RESULTS

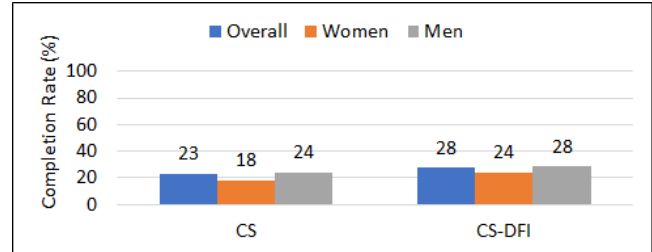
5.1 CS and CS-DFI Course Completion Rates

We first looked at the overall completion rates for paid learners in the CS and CS-DFI courses and broken down by gender (per dataset 1). These results, provided in Figure 1, include twelve CS courses and four CS-DFI courses. The completion rates are calculated by dividing the total number of students who completed the courses by the total number who enrolled; this is done for both CS and CS-DFI courses. The aggregated population totals and breakdowns by gender can be seen in Table 4. This shows CS-DFI succeeded at recruiting a higher percentage of women learners (18.2%) than the CS courses (15.0%).

The completion rates are higher for the CS-DFI course overall and for each gender individually. To determine if these differences are statistically significant, we ran a two proportion z-test comparing the aggregated overall, women and men student populations in CS compared to CS-DFI courses. In all cases the difference is statistically

Table 4: Paid enrollments and gender representation.

Course Type	Enrolled	% Women	% Men
CS	105,364	14.97%	84.15%
CS-DFI	44,292	18.18%	80.82%
OOP	27,288	18.84%	80.23%
DS	8,138	17.50%	81.51%
ADS	5,670	15.20%	83.49%
SE Interview	3,196	19.52%	79.38%

**Figure 1: Completion rates by course type.**

significant (overall: $z = 18.14$, $p < 0.01$; women: $z = 10.99$, $p < 0.01$; men: $z = 15.65$, $p < 0.01$). The gap between women and men student completion rates is also smaller for the CS-DFI courses at four percentage points compared to six percentage points for the CS courses, although in both CS and CS-DFI the difference in gender completion rates is statistically significant (CS-DFI: $z = 7.52$, $p < 0.01$; CS: $z = 16.24$, $p < 0.01$).

Our courses, like some of the other CS courses offered on this platform by our institution, are organized in series. We hypothesized that completion rates may be lower for earlier courses rather than later courses in a series, as students in earlier courses may be exploring whether this is the right course series for them. To examine this hypothesis, we compare the CS-DFI courses to CS courses that are part of a series to see if there is a difference in enrollment based on where courses lay in a series. The results of comparing the CS-DFI courses with courses in two other series of CS courses (e.g., OOP against the 1st courses in CS) appear in Figure 2. We find that course completion rates in general are lower for the first course in the series for both CS and CS-DFI. We also find that CS-DFI completion rates are higher for the first and second course relative to CS, but lower/comparable to the CS series for the third and fourth courses.

We find that women have a lower completion rate than men for all CS courses and our own CS-DFI courses. Encouragingly, the difference in completion rates for women is no larger in the CS-DFI courses than the CS courses, and in 3 of the 4 courses it is smaller.

5.2 CS-DFI Video Engagement

For the purposes of further analyzing the CS-DFI courses with the goal of understanding student participation in the courses, we chose to focus on whether or not students watched the videos provided as content for the course. As discussed in Section 3, each CS-DFI course has different types of videos. By examining dataset 2, we aim to better understand student engagement in the course to see which videos are more popular and if there are differences in watching patterns by gender. Students must have played an entire single video for the system to count it as watched.

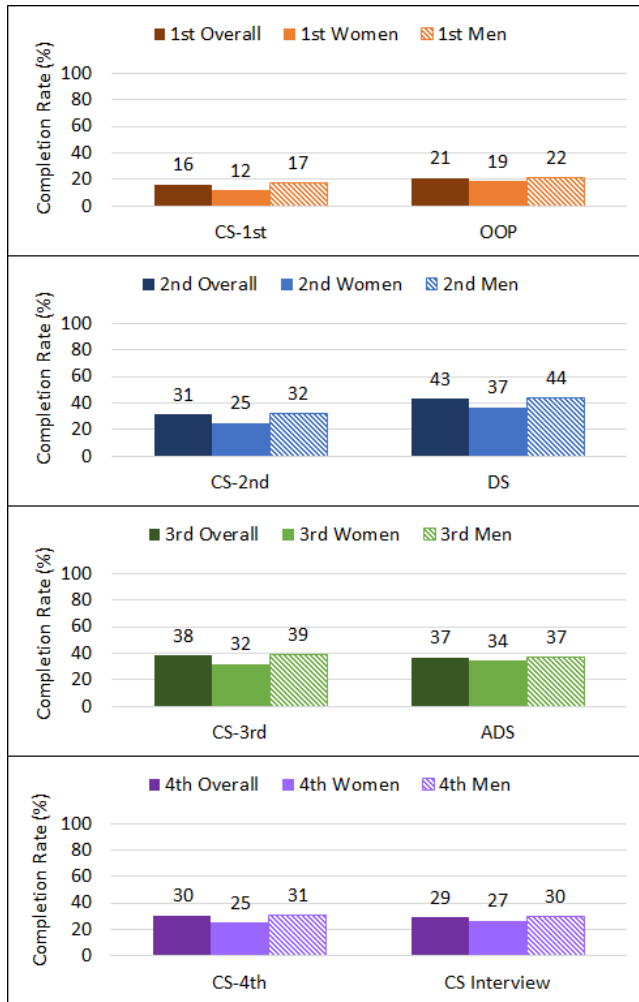


Figure 2: Completion rates by # in series.

The results appear in Figure 3. In all video categories, except for the “In the Real World” (IRW) videos in the DS and IRW and project videos in the ADS course, we find that women watch more of the videos than men. This suggests that women spend more time consuming the videos in the course than men. Interestingly, in the first course of the series (OOP) there is a more even distribution in the types of videos watched, with the project videos being the most heavily watched whereas in the later courses, content videos are watched at a higher rate than the other types of videos. It may be that after students take the first course they are less interested in the other types of videos.

6 DISCUSSION

Completion Rates. Examining our courses against the other CS MOOC offerings from our institution, we are pleased to find that our courses have higher completion rates for both women and men and that the gap between women and men is smaller. In addition, we find that if we compare courses based on their ordering in a series, the percentage of women completing our courses is higher than for all comparable CS MOOC offerings from our institution. Although we aspired to eliminate the gap between men and women and to have even higher completion rates, these results show progress

Table 5: Students’ reason for taking the course.

Answer	OOP (N = 344)	DS (N = 2947)	ADS (N = 1511)
“To brush up on material I already learned”	50%	60%	54%
“To learn this material for the first time”	30%	50%	63%
“To learn Java”	76%	70%	64%
“To complete the specialization”	53%	66%	66%
“To earn the course certificate”	49%	60%	65%
“Some other reason”	4%	18%	18%
“I’m not sure. I’m just browsing”	2%	15%	15%

for student outcomes in our online courses. Lastly, although our data limited our ability to examine outcomes for different racial and ethnic demographic groups, we hope future work will look at this important topic.

Enrollments. A true success for our courses would be to build a MOOC that reaches a more diverse population and which facilitates learner mastery in core CS concepts by emulating practices for inclusive learning from the face-to-face classroom. A key component of such a project is to attract these learners to the courses in the first place. We have some signal of success in this regard in that we found a higher proportion of paid learners were women in our courses relative to the other CS MOOC offerings from our institution. However, it may still be the case that the groups of students we had in mind when designing the courses simply are not taking online courses. This question of recruitment was not addressed in this project and may have significant bearing on the future evaluation of the courses. We encourage future MOOC designers to be sure to build a recruitment plan for their courses.

Learner Feedback. We are pleased to find that learner feedback often reflected back the components we baked into the course design. A learner in OOP wrote of their value of real-world projects in a meaningful context: “This course was perfect, the over-arching theme of working on the Earthquake application was interesting enough and gave me extra things to try and play around with”. In DS, a learner showed us we had succeeded in normalizing struggle when they wrote “...timely videos on how people struggled while learning this stuff knowing that I am not alone [were] just that i need to work hard to succeed”. In ADS, a learner shared their appreciation for the Concept Challenges: “I like the section which is the students discuss together and they share their ideas. I was curious how good students think of a question. Then I found we both have same thinking, if I really study hard”.

Student Goals. In this analysis, we use course completion rates to measure success, despite the limitations of this metric discussed in the literature [5, 7, 10, 15]. Indeed, we find that learners chose to take the CS-DFI courses for multiple reasons and many did not have the same goals. In an optional pre-survey for the OOP, DS and ADS courses, learners were asked, “What is/are your reasons for taking this course? We recognize that you probably don’t know exactly what this course is about, specifically, but take your best guess. Please select all that apply”. Table 5 shows the results of student responses from the three courses. Learners who checked all options on the survey are excluded from these results as it would

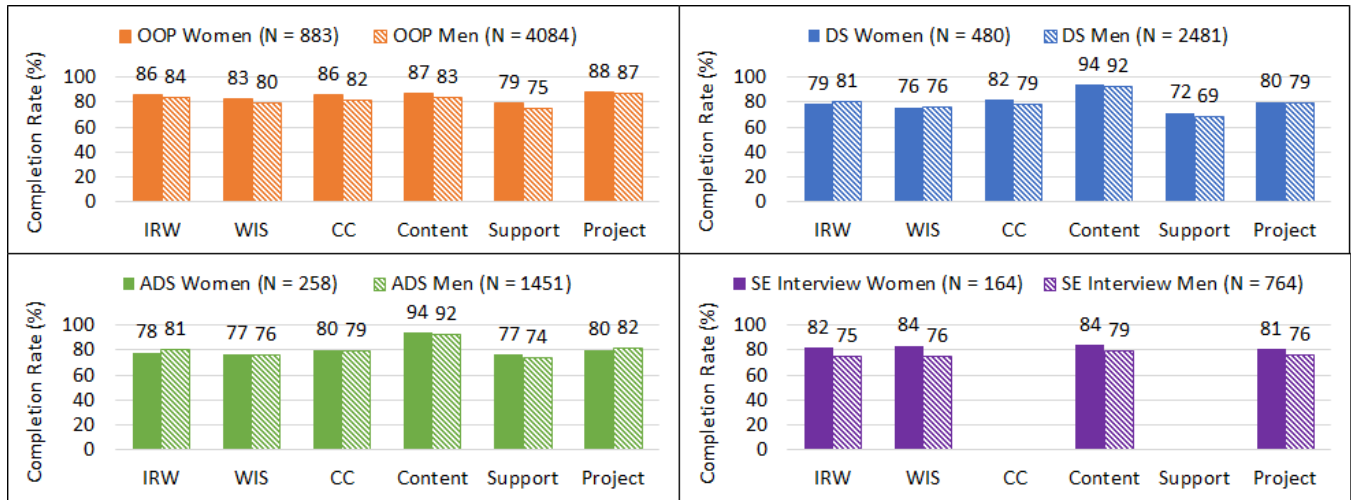


Figure 3: Percentage of videos watched per course broken down by video categories.

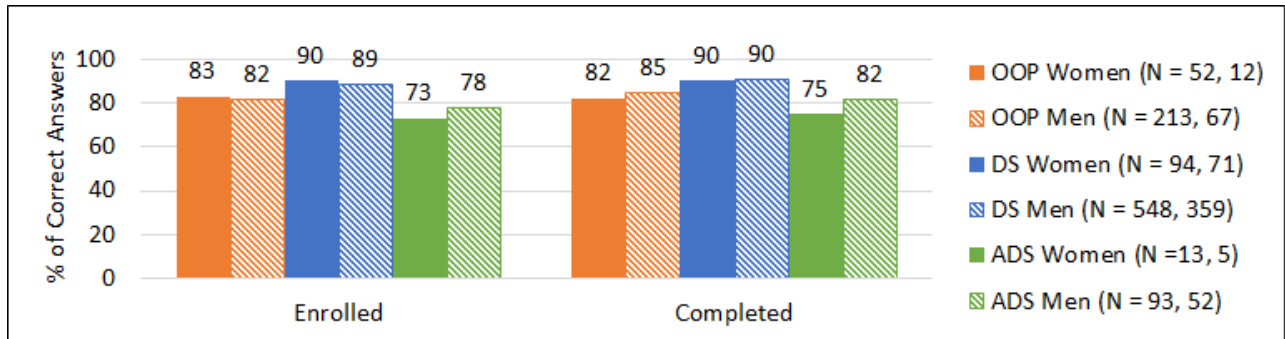


Figure 4: Percentage of correct answers on pre-quiz tested knowledge questions per course.

not make sense for all answers to be true. Looking at the limited responses, there were indeed a wide range of goals among learners. **Student Backgrounds.** There is also variance in the background knowledge of learners as shown in Figure 4. An optional quiz on the course material was given to learners at the beginning of the OOP, DS and ADS courses. This quiz was designed by instructors as part of learning about their students and evaluating if the course was working well for them (see [1] for full quizzes). Response rates were quite low and represent a small fraction of the students in the course, so we hesitate to draw too much from the results. However, the results show that on the pre-class quiz women scored higher than men on the quiz in the DS course and men scored higher than women for OOP and ADS (albeit with tiny sample sizes). Interestingly, looking at only the students who completed the course we do not see a large difference in scores but do see that women who completed the OOP course performed worse on the pre-class quiz than men.

Limitations. Gaps in data pose a persistent threat to validity in the analysis. Our results on gender data are limited to students that provided their gender or for whom gender was inferred by the online platform. The optional pre-quiz and pre-survey suffered from low participation rates. The results for CS-DFI are comparing against other CS MOOCs offered by our institution on the same online platform, but it could be that there are differences in the content of these courses that impact the findings. Lastly, a lack

of published outcomes for men and women in computing MOOC courses limit our ability to make comparisons, although we hope the results we share here can facilitate more research in this area.

7 CONCLUSION

We report on our experiences designing online computing courses to include best-practices for inclusive learning from in-person classes. We compare student completion rates between other computer science MOOCs offered by our university to the computing courses we designed for inclusion and find that our courses have a higher percentage of women enrolled, slightly higher completion rates, and a smaller gap in completion rates between men and women. Examining behavior within the courses, we find that women generally use the available video resources more than men. Also, similar to prior studies, we find that our learners enroll in our courses with a variety of goals in mind. Our experiences designing and evaluating the courses suggest that more work needs to be done to understand how best to recruit diverse students and how to best provide an inclusive environment in online course settings.

8 ACKNOWLEDGMENTS

We thank Marshall Seid and Jason Liu for their early work examining the data on this project as well as Coursera for providing data used in this study. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1650112 and NSF Award #1712508.

REFERENCES

- [1] 2021. Student Background Quizzes from this paper. <https://researchlinks.page.link/mooc-experience-report>. Provided for replication and other use.
- [2] Tucker Balch. 2013. MOOC Student Demographics (Spring 2013). *The Augmented Trader* (27 Jan 2013). <https://augmentedtrader.com/2013/01/27/mooc-student-demographics/>
- [3] Lori Breslow, David Pritchard, Jennifer DeBoer, Glenda Stump, Andrew Ho, and Daniel Seaton. 2013. Studying Learning in the Worldwide Classroom: Research into edX's First MOOC. *Research in Practice and Assessment* (2013).
- [4] Catherine H Crouch and Eric Mazur. 2001. Peer instruction: Ten years of experience and results. *American journal of physics* 69, 9 (2001), 970–977.
- [5] Jennifer DeBoer, Andrew D. Ho, Glenda S. Stump, and Lori Breslow. 2014. Changing “Course”: Reconceptualizing Educational Variables for Massive Open Online Courses. *Educational Researcher* 43, 2 (2014), 74–84.
- [6] Sally Fincher and Marian Petre. 1998. Project-based learning practices in computer science education. In *FIE'98, 28th Annual Frontiers in Education Conference. Moving from 'Teacher-Centered' to 'Learner-Centered' Education. Conference Proceedings (Cat. No. 98CH36214)*, Vol. 3. 1185–1191.
- [7] Philip J. Guo and Katharina Reinecke. 2014. Demographic Differences in How Students Navigate through MOOCs. In *Proceedings of the First ACM Conference on Learning @ Scale Conference (L@S '14)*. 21–30.
- [8] Mark Guzdial. 2007. Contextualized Computing Education Increasing Retention by Making Computing Relevant. (2007).
- [9] Mark Guzdial. 2013. Exploring Hypotheses about Media Computation (ICER '13). 19–26.
- [10] Khe Foon Hew and Wing Sum Cheung. 2014. Students' and instructors' use of massive open online courses (MOOCs): Motivations and challenges. *Educational Research Review* 12 (2014), 45–58.
- [11] René F. Kizilcec, Chris Piech, and Emily Schneider. 2013. Deconstructing Disengagement: Analyzing Learner Subpopulations in Massive Open Online Courses. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge (LAK '13)*. 170–179.
- [12] Joseph S Krajcik and Phyllis C Blumenfeld. 2006. *Project-based learning*.
- [13] Juho Leinonen, Petri Ithantola, Antti Leinonen, Henrik Nygren, Jaakko Kurhila, Matti Luukkainen, and Arto Hellas. 2019. Admitting Students through an Open Online Course in Programming: A Multi-Year Analysis of Study Success. In *Proceedings of the 2019 ACM Conference on International Computing Education Research (ICER '19)*. 279–287.
- [14] Colleen Lewis, Paul Bruno, Jonathan Raygoza, and Julia Wang. 2019. Alignment of Goals and Perceptions of Computing Predicts Students' Sense of Belonging in Computing (ICER '19). 11–19.
- [15] Allison Littlejohn, Nina Hood, Colin Milligan, and Paige Mustain. 2015. Learning in MOOCs: Motivations and self-regulated learning in MOOCs. *The Internet and Higher Education* 29 (2015), 40–48.
- [16] Leo Porter, Cynthia Bailey Lee, and Beth Simon. 2013. Halving Fail Rates Using Peer Instruction: A Study of Four Computer Science Courses. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. 177–182.
- [17] Leo Porter, Cynthia Bailey Lee, Beth Simon, and Daniel Zingaro. 2011. Peer instruction: do students really learn from peer discussion in computing?. In *Proceedings of the seventh international workshop on Computing education research*. 45–52.
- [18] Julie Posselt. 2018. Normalizing struggle: Dimensions of faculty support for doctoral students and implications for persistence and well-being. *The Journal of Higher Education* 89, 6 (2018), 988–1013.
- [19] Robert Pucher and Martin Lehner. 2011. Project based learning in computer science—a review of more than 500 projects. *Procedia-Social and Behavioral Sciences* 29 (2011), 1561–1566.
- [20] E Rivers. 2017. Women, minorities, and persons with disabilities in science and engineering. *National Science Foundation* (2017).
- [21] Adrian Salguero, Julian McAuley, Beth Simon, and Leo Porter. 2020. A Longitudinal Evaluation of a Best Practices CS1. In *Proceedings of the 2020 ACM Conference on International Computing Education Research (ICER '20)*. 182–193.
- [22] Beth Simon, Julian Parris, and Jaime Spacco. 2013. How We Teach Impacts Student Learning: Peer Instruction vs. Lecture in CS0. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. 41–46.
- [23] Paul Tough. 2014. Who Gets to Graduate? *The New York Times Magazine* (15 May 2014). <https://www.nytimes.com/2014/05/18/magazine/who-gets-to-graduate.html>
- [24] Brenda Cantwell Wilson and Sharon Shrock. 2001. Contributing to success in an introductory computer science course: a study of twelve factors. *Acm sigcse bulletin* 33, 1 (2001), 184–188.
- [25] Daniel Zingaro. 2014. Peer Instruction Contributes to Self-Efficacy in CS1. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE '14)*. 373–378.