



DOI:10.1145/3503915

Overcoming the inherent challenges.

BY UMAKISHORE RAMACHANDRAN AND ZVI GALIL

Creating a Revolutionary Academic Program

THE UNIQUE BLEND of quality education, research, entrepreneurship, and economic development embodied in the operational model of academic institutions in the U.S. is unparalleled in the world. This unique model has been powered by the intellectual commitment and academic freedom of the faculty. Therein lies the rub for creating new and revolutionary academic programs. This article chronicles the creation of the Georgia Tech (GT) Online Master's in Computer Science (OMSCS) program, which is based on massive open online course (MOOC) technology. By relating our experiences—discussing the creative solutions we came up with as well as how we overcame the challenges we faced—we hope we can help colleagues and peers embarking on similar endeavors.

Supply and Demand for Skilled IT Professionals with a Graduate Degree in CS

In the early 2010s, we witnessed a significant uptick in the applicant pool for the MS program in Computer Science (CS) at GT. This mirrored a nationwide trend that was a response to the need for qualified professionals with an advanced skillset to meet the workforce market demand. While half the applicants had academic records worthy of being admitted to the program, the program could admit less than 20% of those who qualified due to limitations on classroom capacity, faculty teaching load, teaching assistant (TA) availability, and other logistical reasons. We needed to figure out a way to scale up MS enrollment while being cognizant of various challenges.

Advent of MOOC. Advances in Internet connectivity have leveled out access to digital content across the globe.² Just as business process outsourcing (BPO) took off thanks to global Internet connectivity, so did online education—specifically, the concept of MOOCs. Individual faculty at several premier institutions spanning a variety of scientific and engineering fields offered free online courses—from introductory to graduate-level—open to anyone. Faculty recognized that making such courses accessible to a global audience represented not only a valuable service to the community but also a potential business opportunity.

» key insights

- **Creating a revolutionary online academic program in institutions of higher learning is fraught with challenges, since faculty are loathe to see the reputation of the institution sullied by any missteps.**
- **The pathway to success combines several key steps: create an environment where faculty can freely exchange ideas and concerns; address the scalability of admission, student credentialing, and exam-proctoring processes from the beginning; devise strategies for continuous monitoring and quality control; and design flexible mechanisms to accommodate the diverse needs of the online student community.**

ILLUSTRATION BY CHARIS TSEVIS



Coursera and Udacity were the first to offer platforms for delivering streaming course content at scale.

Enrollments exceeded 100K globally, especially in popular CS courses like machine learning. Yet the single-digit-percentage completion rate for most courses implied some difficulties.⁴ To reduce attrition, the MOOC model needs to reach a target audience that is motivated and academically prepared, and the model must provide widely acceptable credentials upon completion.

Using the MOOC Model to Bridge the Supply/Demand Gap of Skilled IT Professionals

The College of Computing (CoC) at GT saw an opportunity to use the MOOC model to close the supply/demand gap for skilled professionals. CoC envisioned a quality CS graduate program enrolling thousands of students from the U.S. and beyond, with minimal disruption to their everyday lives and at a fraction of the tuition paid for on-campus programs. The partnership between CoC and Udacity began in 2012 based on this vision. Beyond the challenges associated with creating the program, institutional administrators at many levels needed to be convinced of its academic and economic viability.³ This article focuses on the pedagogical challenges associated with creating this program.

From Vision to Implementation

Creating a blueprint for the program while keeping faculty in the loop proved to be the first hurdle. The second hurdle was proselytizing the program's benefits to fellow faculty and convincing them that it will help them reduce their workload in the long run.

While MOOC platforms such as Coursera, Udacity, and EdX make it possible to disseminate course content worldwide, admitting students at scale and credentialing and assessing their performance with the same rigor and expectations of on-campus programs requires some serious thought. Additional challenges include making the program affordable, contingency plans should the program need to be closed, and timely course production.

The rest of this section identifies program challenges and discusses solutions devised by CoC faculty and ad-



To reduce attrition, the MOOC model needs to reach a target audience that is both motivated and academically prepared.



ministration in roughly eight months to launch the program in January 2014.

Challenge 1: Faculty buy-in. Academic programs are created by faculty, not by administrative decree. The first and foremost challenge is to anticipate and address any justifiable concerns faculty may have, including ensuring that the institute's reputation is not sullied in any way and that the administration's expectations are transparent. The right way to create a successful academic program is to float an idea and let the faculty debate its pros and cons. In the end, faculty should take ownership of the program to feel committed to its successful launch and long-term sustenance.

Creating an environment for free exchange of ideas and concerns. In the fall of 2012, a faculty-based committee was created to study the pros and cons. The committee, which by design had no administrators, consisted of Tucker Balch, Frank Dellaert, Nick Feamster, Jim Foley, Ayanna Howard, Guy Lebanon, Alex Orso, Kishore Ramachandran (chair), Patrick Traynor, Rich Vuduc, and Bob Waters. The committee's charge was to set the table for the free and open exchange of ideas, both for and against the proposed new program, with the entire faculty. Student representatives were also included on the committee to get their perspectives. Committee meetings were open to faculty participation.

The atmosphere in the committee was almost akin to drawing up an action plan for a technology startup. A SWOT ("strength," "weakness," "opportunities," and "threats") analysis was conducted and presented to the faculty in early fall of 2012. Through a series of intense bi-weekly town hall meetings conducted through the fall, several thorny issues were ironed out, including incentivizing faculty for participating in the program, admission requirements, credentialing students, computational requirements, contingency planning for students enrolled in the program, and more. There was also concern about the demographic mix that would benefit from the program and how we could increase access to the program for a diverse mix of students.

Through the following semester of active planning and engaging faculty,

the official plan was presented to the faculty for a vote in March 2013. Since the faculty was engaged in the process from the very beginning, a formal vote resulted in more than 75% of the faculty voting to approve the program.

Specific steps were taken in the proposal to address the challenges related to faculty buy-in for the program.

Incentivizing faculty participation. Academic freedom is the cornerstone of American institutions of higher learning. Indeed, faculty members are entrepreneurs in their own right. They conceptualize research ideas, seek out external funding, make intellectual connections within and across institutions, advance curriculums within their sphere of influence, and so on. Consequently, they are under pressure to simultaneously achieve their personal aspirations while fulfilling their professional commitments to the institutions they serve.

Therefore, the committee decided from the very start that participation in the new program should be optional. That is, there would be no administration-mandated increase in faculty workload. Faculty participation to support the program's creation, delivery, and long-term sustenance was to be entirely on an opt-in basis. But this also created an interesting dilemma: We needed a significant fraction of the faculty to produce the courses to provide a healthy set of options for students. How would we motivate the faculty to opt in?

Fortunately, we had some prior experience in this regard. In 2007, we won a four-year contract with the South Korean government to offer an MS program in embedded software (partly streamed from GT and partly by GT faculty teaching in person at partnering Korea University in Seoul). To incentivize faculty to participate in this program, which was outside their normal workload, they were offered extra compensation beyond their normal academic salary.

The committee devised a similar incentive plan for compensating faculty to develop video course content for the MOOC-based OMSCS program. Producing video course content would be akin to writing a book, with each faculty member receiving a one-time compensation of 30,000 USD and GT re-

taining the course video copyright. Additionally, every time the course video is subsequently used, the faculty member receives a "royalty" of 2,500 USD.^a The faculty member who develops the course is expected to be the instructor on record the first time the course is offered. Subsequent offerings of the course could be performed by any faculty qualified to teach the material. However, the course developer very often continues to be the instructor in subsequent semesters as well.

Also, the faculty member who runs the course during a semester gets extra compensation of 10,000 USD for that service. Courses were to be run asynchronously with the expectation that students watch the recorded video lectures. The faculty member who runs the course is responsible for administering exams and projects, interacting with students on public forums, and assessing student performance. We deemed this workload to be at most eight hours a week (roughly equivalent to a one-day-per-week outside consulting opportunity that is available to any faculty member at GT). It is on this basis that we arrived at the compensation model for running the course.

The extra compensation idea, socialized with the faculty during the town hall meeting, received a positive reception. More importantly, the faculty saw another, albeit unplanned, positive consequence to producing video lectures of their course content. Minimally, it would liberate them from having to find substitute help to cover course material while away when the same course was offered on campus. More adventurous faculty also saw an opportunity to flip the classroom—scheduled teaching time could be used for enriching discussions by having students preview the video lectures ahead of time.

Getting the nod from faculty and upper administration. In March of 2013, the faculty voted to approve the program. This was the first step. Next, we had to secure approval from GT administrators at the highest levels. Every university may have its own internal approval processes, so the specific

steps we had to take may not directly apply exactly to other institutions. Yet, for the sake of completeness, we mention our internal process.

This program was the first in GT's history where all the courses to satisfy a degree requirement were to be offered in distance-learning format. In 2007, we secured approval from GT administration to create the joint GT-Korea University MS program in Embedded Software we alluded to earlier, in which 50% of the courses were taught via distance learning. We were able to ride the coattails of that experience to get the nod from upper administration to create the new program. Subsequently, GT administration secured the program's approval with the Board of Regents of the University System of Georgia.

Challenge II: Creating an online education program at scale. The aspirational goal of the program was to offer a MOOC-based MS degree at scale with the same rigor, rights, and privileges as the on-campus master's program at a fraction of the cost. This aspirational goal raised several thorny issues the committee had to iron out.

Admission criteria. The committee wanted the program to be accessible to knowledge seekers primarily in the U.S. but available to anyone in the world who met acceptable academic standards. The minimum requirement was deemed to be an undergraduate college degree, though not necessarily in CS. Beyond that, how do you judge if a student is adequately prepared for the program? It became clear that traditional admission criteria, such as GRE and letters of reference, may not work to implement admissions at scale. The committee devised a novel admission criterion for the steady state: Let aspiring students sign up and pay for two of the MOOC-based courses developed for the program. If they perform well, they are admitted into the program and the courses they already took count toward their degree.

Credentialing students and proctoring exams. Administering programming assignments and projects for a graduate degree in CS through a learning management system (LMS) is quite natural since student submissions are done online even for on-campus offerings. Exams are a different matter. Students could be enrolled in the program

^a Even if faculty members leave GT, they continue to receive royalty payments every time the course is offered in the program.

from potentially anywhere in the world. We could not mandate a fixed time for everyone to take the exams. Also, credentialing students and remote proctoring are important challenges that could threaten the program's integrity if not handled correctly.

One possibility was to establish satellite centers, where students could take the exams with on-site physical proctoring. Unfortunately, this solution does not scale and would make the program prohibitively expensive, which is at odds with the high-level goal we set for ourselves. Thanks to advances in computer vision, camera-based remote proctoring obviated the need for manual vigilance of the camera streams. Start-ups were beginning to offer these types of services;⁶ we could work out the details of administering remote exams at scale using such a platform. The specific platform we decided to use, Proctortrack,^{6,b} allows exams to be created and administered in a video-based proctoring environment that ensures integrity.

Flexibility for knowledge seekers. The expectation is that this program would attract knowledge seekers who are at a different stage in their lives—specifically, students who are already in the workforce and see this as an opportunity to enhance their skillset without a major disruption to their lives. We also understand that life happens; students in the program may start, stop, and take longer to earn a degree while juggling, for example, a day job and the course work. We built flexibility into the program to allow students to drop a course without it affecting their GPA and/or to withdraw from all courses in a semester without any repercussions on their transcripts.

Challenge III: Aligning online and on-campus offerings. One of the guiding principles in developing the program was ensuring that the online program would have the same rigor and expectations as on-campus courses. In this sense, except for the delivery format, the learning outcomes for every course had to be the same as its on-campus counterpart.

Maintaining rigor and meeting expectations. Students who sign up for

the program could come from varied backgrounds. They may not even have an undergraduate degree in CS. This is precisely the reason for the admission criterion we used, which was based not on their backgrounds but on their ability to do well in a couple of foundational courses before they are formally admitted into the program. It helps knowledge-seekers self-assess their preparedness and succeed in the program.

To put this challenge in its proper perspective, misalignment of course expectations and student preparedness also happens in on-campus graduate programs. The usual remedy is to direct students to either audit or take an undergraduate course as remedial work to prepare for the graduate course. Unfortunately, this mechanism is not viable for online students since undergraduate courses are not available in online format. Instead, for every graduate course, we fully disclose the assumed knowledge units and skillsets (in terms of programming, mathematical, and other proficiencies) so that there are no surprises for students taking a graduate course. We point to online resources (for example, both at GT and publicly available MOOC courses) that students may use to prepare for any given graduate course. We have also been creating and offering introductory online courses to aid potential students in acquiring the necessary background. For example, we have created an introductory Operating Systems (OS) course to prepare students for the more rigorous graduate-level OS course and three undergraduate programming courses.

A natural question that arises is about the disparity in the educational experience of online students compared to on-campus students. On-campus students have the advantage of in-person interaction with faculty. However, that advantage diminishes rapidly beyond the first few rows with class sizes larger than 75 students. In fact, on many campuses, in popular entry-level courses (for example, machine learning), it is not uncommon for students to sit in overflow areas and watch lectures on overhead screens, since the class size (greater than 100) far exceeds classroom capacity. Of course, on-campus students can visit professors during

office hours to establish a rapport that is not possible for the online students. Nevertheless, enterprising online students find ways to establish personal relationships with professors; many of them supplement a course they like by taking independent study with the professors. On the other hand, it is also worth mentioning that only a small fraction of on-campus students establishes personal rapport with faculty.

Computational resources for programming projects. One would be tempted to think that provisioning computational resources globally to carry out CS projects in the courses should not pose any serious challenges thanks to cloud technology. As an aside, even with on-campus courses there is a trend toward relying on the cloud to meet computational needs. However, some courses may have specialized needs. For example, there may be a need to access the bare metal for accurate timing measurements in some programming projects. This would not be possible due to the virtualized nature of the cloud resources. There is no easy solution to this problem since it is not feasible to provide access to bare metal at scale locally in any academic institution. Faculty may need to devise creative approaches to circumvent such issues while still providing the expected learning experience. As an example, in the advanced Operating Systems course, students are expected to implement and perform comparative timing measurements of different synchronization algorithms on parallel machines. We opted to use cloud resources to allow students to implement and validate the functional correctness of their algorithms. The instructor provides students with comparative timing measurements of the algorithms on a real parallel machine (implemented by the teaching team), and the students are asked to explain the results.

More generally, moving programming assignments to the cloud requires some initial investment of time for faculty and teaching assistants, as well as a learning curve to overcome to adapt to the cloud. Besides, there are multiple cloud providers, and individual faculty members may prefer one over another. Each institution has a different approach to this issue, but it is

^b For administrative reasons, we recently switched to a different proctoring system called Honorlock.

now bubbling up at the national level as evidenced by the recent roundtable discussion organized by the Computing Research Association (CRA).¹

Calendar alignment. It is also conceivable that a faculty member would be teaching two versions of the course concurrently during the same semester. Cross-fertilization of student experience across these two simultaneous offerings is another enabler to maintain the rigor and expectations. From this perspective, and from the point of academic logistics, we decided that the online courses (start and end) would align with GT's semester schedule. Further, such an alignment with the campus calendar would make it easier for faculty, students, and the teaching team to plan their lives.

Credit transfer. Evaluating courses taken elsewhere for transferring credits is not new for any academic program. In the normal scheme of things, a faculty member with domain expertise would be asked to weigh in on such matters. However, the expected scale of the program could swamp the system if it is not clearly laid out at the outset. The concern was to ensure faculty are not overburdened with such requests. We opted to increase the amount of professional administrative technical staff to deal with the potential increase in this load in a way that is consistent with the practices already in place for dealing with such requests. This resulted in minimal additional burden on the faculty.

Moving between online and on-campus. Some knowledge seekers may not be able to continue the pursuit of the program, even an on-campus program, for a variety of reasons, such as economic conditions, family situations, and so on. By the same token, a student who signs up for the online program may want to move on campus simply to experience campus life. The committee took it upon itself to clearly identify such non-standard pathways for maximum flexibility while adhering to realities, such as on-campus classroom capacities. The process for switching is quite simple. A student in the online program fills out an online form desiring to switch campuses well in advance of the semester start date and makes the academic advisor aware. Requests are usually approved after a



CoC envisioned a quality CS graduate program enrolling thousands of students from the U.S. and beyond.



couple of months and some logistical parameters are considered (for example, impact on on-campus class sizes, applicant's academic progress, and so on). It is uncommon to have requests turned down, but it does happen. Moving in the other direction is similar. To date, the traffic in either direction has been minimal. Between summer 2018 and fall 2021, around 150 students have moved from OMS to on-campus, and around 40 have moved from on-campus to OMS.

Challenge IV: Creating an affordable yet self-sustaining online program. We hoped to create an affordable higher-education program at scale using MOOC technology at a fraction of the cost of quality on-campus programs. We knew we could do it in the steady state. The question was, how would we finance it in the short run? Being a public university, GT has fiscal constraints. In parallel with the deliberations by the faculty committee to iron out the details, CoC administration worked to create the financial base for bootstrapping the program. Specifically, AT&T came on board to provide a generous initial gift of 2 million USD to produce the courses and launch the program pending approval by the faculty. We estimated that we needed 1,000 students in the program to break even at an affordable tuition of less than 7,000 USD per student to complete the degree requirements. We were optimistic that we would get there in a couple of years, but we wanted to start small to mitigate any rookie mistakes. AT&T followed its initial gift with an additional 2 million USD during the ramp-up period, until revenue from the program began covering the costs.

This begs the question, could GT have pulled this off without those timely gifts from AT&T? The answer is a qualified yes. Producing each course is like a mini movie production, and the initial cost estimate of producing each course with Udacity was 300,000 USD, which included the one-time compensation of 30,000 USD for the faculty. However, depending solely on GT to finance this effort would have resulted in a slow start at best. Plus, it might have provided naysayers in the faculty with a stronger voice. Finally, it may have been harder to justify the significant cost reduction for a student in the OMS program

compared to the on-campus program.

The key to fundraising for such an effort is to pitch it as a win-win for all partners. AT&T believed the program could provide its own employees with an opportunity to elevate their skillset, and it would serve as an indirect retention tool for the company. Udacity had a share in the revenue: Initially it was 40%, later it became 35%, and now it is 0% since the program is entirely run from within GT.

To date, we have more than 50 courses produced and offered in the program. Production cost has been reduced to 100,000 USD since it is now done internally using resources from GT's professional education unit instead of through Udacity. Revenue from the program in AY 2020 was 13 million USD. Revenue-sharing is split three ways: GT-central (55%), CoC (35%), and GT professional education (10%).

Challenge V: Exit strategy. A three-way partnership between GT, AT&T, and Udacity was at the core of the program being envisioned. The program would help transform the lives of knowledge seekers who either could not afford the high cost of graduate-level higher education and/or afford the disruption in their lives to enter a full-time on-campus program. At the same time, we had a responsibility to ensure that there would be pathways for students who come on board to finish the program if for any reason GT could not continue to offer the MOOC-based program. There have been several instances of academic programs started by Western institutions in other countries that folded for sundry reasons. Therefore, the committee also worked out the exit-strategy details while ensuring that students already in the program were not left in the lurch. Succinctly put, the exit strategy was a commitment to current enrollees in the program that they would be able to complete their degree requirements if the program offering is canceled for any reason.

Challenge VI: Pipelining course production. Once the approvals were in place, the real work started. We committed to launch the program in January 2014; at the time, we had about eight months to get a set of courses ready for the launch. This aggressive timeline would have been impossible to meet if we had waited for all the program courses to be produced before the launch date. Another huge bottleneck,



Moving programming assignments to the cloud requires some initial investment of time for faculty and teaching assistants.



apart from faculty availability, was the limited facilities available for recording videos in a short amount of time.

The approach we took was to pipeline course production. A small coterie of faculty—Nick Feamster (Networking), Charles Isbell and Michael Littman (Machine Learning), Alex Orso (Software Engineering), Sebastian Thrun (Robotics), and Kishore Ramachandran (Advanced Operating Systems)—pioneered video content creation for the first set of five courses we used to launch the program. In parallel, we got commitments from additional faculty who opted in to produce courses so that we could keep the course production pipeline busy and have courses ready for subsequent semesters.

Udacity was the MOOC platform used for course production. Udacity had a unique pedagogical style that gives students the feeling of interacting one-on-one with an instructor, akin to sitting at a table and sketching out ideas on a piece of paper with a friend. This meant that producing a course video was not as simple as narrating over an existing PowerPoint. It required faculty to rethink how to make the student's experience "personal" despite being virtual. This meant writing presentation content in one's own handwriting. Since the attention span of a learner is typically less than 10 minutes, one had to plan how to break up an hour-long video into roughly six to ten segments. Also, to break the monotony and to help assimilate the content, one had to think about quizzes that could be incorporated between video segments.

Every course was assigned a course developer by Udacity. Though the developer would be conversant with the course's technical content, the developer's primary role was to help the course owner create course material consistent with Udacity's pedagogical style. The time commitment to produce the course was non-trivial. Producing a one-hour video required eight hours of work on average, including actual recording time. Those who signed up to create the first set of courses would admit that they grossly underestimated the effort to produce their course videos. But their initial experience helped to streamline course production in later semesters. For example, personal-

ized fonts matching the writing style of each course owner were developed so that they could type the content rather than having to handwrite them as the pioneers did for the first set of courses.

How Does a Typical Course Run in the OMSCS Program?

As an example of how a course is run, let us review CS 6210, an Advanced Operating Systems (AOS) course taught by the first author. Content for the course is entirely drawn from a set of seminal papers. Apart from the fact that there are no live lectures, student expectations in terms of the assessment units that determine course performance are the same as in the on-campus offering of the same course.

Students are given a weekly schedule of lectures they should watch. The professor engages with students in weekly, one-hour, live video hangouts, reviewing the material from the previous week's videos^c and answering student questions. Students also participate in discussion forums, where they discuss course material, projects, homework, and exams. The video hangout helps to address some of the lingering questions in the online forums through direct, live interaction of the professor with the students.

c Similar to the way courses are taught in an on-campus setting, it is the course owner's responsibility to periodically update content—that is, the recorded videos. The system provides the machinery necessary to enable such updates. Updates have been driven mainly by course owners, except for a complete redo in two courses where the course ownership changed.

Due to time differences, not all students can attend live hangouts, so they are recorded and made available online. The AOS course has four hefty programming projects; students use a LMS^d to access and submit projects. In addition to the weekly hangout with the professor, TAs assigned to the course offer office hours to answer questions regarding projects.

Timed tests are conducted using Proctortrack. Students taking tests must have a webcam on their computer. They begin by showing their student IDs for credentialing, and Proctortrack takes control of their computers for the test's duration. Further, students must show the area where they are taking the test. The only action students should take during the test is to type the answers to the questions displayed on their screen, as well as navigating back and forth through the test.

Proctortrack records video and audio of the student taking the test. Its postmortem analysis flags places in the video that an instructor/TA may need to check for any potential infractions. In our experience, the number of severe warnings is quite small. For example, in a 120-minute test taken by

d Canvas is the LMS that is currently used for the OMSCS program.

100 students, we would typically observe less than 10 severe warnings, such as students switching tabs during the test, opening another browser window, and so on. Most of these warnings usually turn out to be benign and are quickly resolved by the TAs. The number of false positives and false negatives generated by the system is also quite small. Anecdotally, some students have privacy concerns as to how the data collected during proctoring will be used. We assuage student anxiety to some extent via the clearly stated policies of the proctoring company we use. Nevertheless, some students take steps to minimize risk, such as using a non-personal laptop and uninstalling the proctoring software after the test.

Using this course as a sample, we compared the performance of on-campus students with that of students taking the online version of the same course in fall 2020. It should be reiterated that the course content and the graded items are the same for both offerings. Due to COVID-19, the on-campus offering was also online; the only difference is that on-campus students had the advantage of live, synchronous streaming of the lectures. Table 1 shows the distribution of grades for the two offerings. Pre-pandemic, on-campus offerings resulted in a higher

Table 1. Comparison of an exemplar course offered on campus and OMSCS.

Program	Total enrollment	As	Bs	Cs	Ds	Fs
On campus	57	40 (70%)	10 (18%)	3 (5%)	3 (5%)	1 (2%)
OMSCS	160	130 (81%)	19 (12%)	6 (4%)	1 (1%)	4 (2%)

Table 2. Demographics of OMSCS enrollment.

Year	Number of Students	Median Age	Demographics				Demographics		
			Women	Men	U.S.	Intl.	White	Non-White	Asian
Spring 2014	380	33	9%	91%	85%	15%	66%	11%	23%
Spring 2021	11,085	29	20%	80%	55%	45%	36%	14%	50%
Cumulative (2014-2021)	23,839	28	18%	82%	68%	32%	40%	15%	45%

Table 3. Demographics of OMSCS graduates.

Year	Number of Students	Demographics				Demographics		
		Women	Men	U.S.	Intl.	White	Non-White	Asian
Fall 2015 (first batch)	18	Less than 5	More than 13	65%	35%	45%	0%	55%
Cumulative (up to spring 2021)	4,640	15%	85%	70%	30%	47%	12%	41%

percentage of As (for example, 77% in the fall of 2018). Though it is just a sample, it perhaps goes to show that online students were more prepared for remote learning than on-campus students during the pandemic.

We used a separate public forum (Piazza) for each of the online and on-campus offerings. Instructions to both groups of students were the same: Use the forum to discuss any conceptual or project-related questions and for collaborative peer learning. In general, online students tend to be more collaborative, perhaps due to demographics (the median age of OMSCS students tends to be around 30) and the fact that public forums are the only way they can interact with their fellow students.

A Look Back at the Program Since Its Launch

We announced the OMSCS program in the summer of 2013 to invite applicants, and it launched in the spring of 2014 with an initial enrollment of 380 students. While the original intent of the MOOC approach was to educate vast numbers of learners at once, we initially kept the numbers small—for all the right reasons (admission logistics, credentialing, teaching-team size, testing to ensure program integrity, and so on). Our intent was to start slow, fix any kinks in the system, and ramp up in subsequent semesters. The launch garnered considerable media attention due to the program's ability to offer higher education at a fraction of the cost of on-campus education. Plus, OMSCS makes it possible for knowledge seekers to retool their skill-sets without disrupting their lives. Post-launch, we had concerns about the many things that could potentially go wrong. But thanks to the faculty's untiring pre-launch efforts, there were only tiny hiccups, such as not knowing the scalability of some public-utility software for document sharing, which was used for team formation in group projects. We were pleasantly surprised that there were no catastrophic events to torpedo the program. Still, the first few semesters were tense, despite the support we received from our peers both nationally and across the globe.

In December 2015, after just six semesters (including the summer semesters), we had the first OMSCS program

commencement. We were amazed to see that about 18 students, most with full-time jobs and families, had completed the program requirements in such a short time despite the demands and pressures in their lives. Watching this first group of graduates receive degrees was emotional; without the OMSCS program, many of them could only dream of such an achievement due to their respective circumstances.

As of this writing, the official OMSCS enrollment had grown to 11,085 by the spring of 2021, and a total of 4,640 students have graduated (not including spring of 2021 graduates) from the program. Table 2 provides the demographic breakdown of student enrollment. The demographic shift is interesting to note. In the first intake of students in 2014, 66% of the 380 students identified as White Caucasian, 23% as Asian, and 11% as non-White (Black, Hispanic, and so on). In the spring of 2021, with an enrollment of 11,085, the White Caucasian demographic accounts for only 37%, 50% for Asian, and 13% non-White. It is also a welcome trend to see that the number of women enrolled in the program has grown from 9% initially to 20% in the spring of 2021. For comparison, in the spring of 2021, the number of women enrolled in the on-campus MS program in CS at GT was 24%. The higher percentage of women in the on-campus program could be attributed to the fact that most of them are international—primarily from India, China, and South Korea. On the other hand, most of the women in the online program are domestic students. It is well known that low enrollment of women in STEM programs, especially in CS, is a major problem in the U.S. and there are concerted efforts to remedy this problem—for example, broadening participation in computing by the National Science Foundation. International enrollment in the program has gone up from 15% initially to 45% in the spring of 2021.

Table 3 offers demographic information for students who have graduated from the program to date. The cumulative number of women graduates has grown to 15% from a very small number in the first batch. The cumulative number for non-White graduates is at 12% (excluding international students), which would mostly comprise ethnic minorities (Blacks and Hispan-

ics). This number is not much different than the on-campus graduate program in CS at GT or elsewhere in the U.S.

Prospects after earning a degree. Diplomas awarded to OMSCS graduates are the same as the ones on-campus graduates receive. Therefore, an employer cannot distinguish between on-campus and online graduates. A survey of OMSCS graduates conducted in the spring of 2021 (351 students participated) revealed some interesting statistics about the program:

- ▶ 96% said the program was worth the investment.
- ▶ 95% said they would recommend it to others.
- ▶ 81% said it helped their careers.
- ▶ 48% said it helped them secure a higher salary.
- ▶ 36% have joined a new workplace since completing OMSCS.
- ▶ 25% have been promoted since completing OMSCS.
- ▶ 6% have started teaching CS at either high school or college levels.
- ▶ 5% have transitioned into the tech sector from outside of it.

Unplanned consequences. A boost in student self-esteem and confidence has been one of the program's most significant unplanned consequences—at least according to anecdotal evidence gathered from personal stories at commencement and/or public forums such as Reddit. For example, many students say they never would have been able to earn an advanced degree from an institution of higher learning such as GT. Some openly admit that, given their credentials, they did not think that they had a chance to enter the program when they applied. Our admission criterion is based on what students are currently capable of, not what they did not accomplish in the past. While pragmatics dictated our admission criterion, it is heartening to see this unplanned consequence of our decision.

Another unplanned consequence is the mentoring and support that OMS alums and seniors freely offer to new entrants. This lifts much of the advising and counseling duties off the shoulders of the OMSCS workforce so they can focus on improving the program and scaling up enrollment.

A further unplanned consequence is the eagerness of OMSCS alums and

seniors to give back to the program. Many volunteer to be TAs for the courses they enjoyed, sometimes without compensation. With the program's current scale and what it is expected to become, it would be nearly impossible to offer the courses and depend solely on the help of on-campus TAs.

Impact on on-campus MS program.

There were some well-grounded concerns on the part of faculty that OMSCS could put a big dent in the on-campus MS program. Fortunately, this did not happen for a few of reasons. For one thing, the on-campus program offers students opportunities to personally interact with faculty. This was crucial to placing students in valuable internships and more; entry-level graduate students (especially from abroad) recognized this value proposition. Further, the OMSCS program vastly increased the pool of required TAs, which meant that most MS students received financial support. Lastly, a significant fraction of entry-level graduate students wanted to test their own passion for doctoral studies by first enrolling in the on-campus MS program. In addition, many international students take the on-campus program so they can get an entry visa to the U.S. It is interesting that for these reasons, on-campus enrollment has shown no decline, though it has not grown significantly, mostly due to our capacity limitations.

OMSCS program logistics. Typically, OMSCS students sign up for one or two courses every semester. The average is 1.3 courses per student. Enrollment attrition does occur in individual courses every semester, the most likely reason being that students do not have the right background for the course. However, it could also be attributed to family and work circumstances. An interesting evolution in this regard is how much new entrants heed the advice of the seniors. The AOS course referenced earlier is a case in point. Despite abundant information available to students on the required background to sign up for this course, the attrition rate reached as high as 50% during the first couple of years. However, this number has stabilized to a more predictable 10% to 20%, which is similar to the on-campus enrollment attrition rate for the same course. We believe the primary reason is the wisdom imparted to as-



Faculty should take ownership of the program to feel committed to its successful launch and long-term sustenance.



piring new students by those who have successfully completed the course. Note that a student who drops a course in one semester often comes back to take the same course the next semester. The average attrition rate per course for the entire program is around 6%.

Admission into the OMSCS program was trickier than we had originally envisioned. Initially, the idea of allowing students to prove they belonged in the program by taking two foundational graduate courses as non-degree-seeking students seemed like a good strategy. However, this proved to be more daunting logistically as we scaled up. To that end, we've slightly modified the process. Now, all students who meet the minimum admission criteria^e are accepted "conditionally." Within the first year, they must complete the two foundational graduate courses and earn a 'B' or better to be formally accepted into the program.

The program's retention rate—that is, students who have either graduated or are still in the program—is 65% from its inception. The 35% who do not make it to the finish line includes those who did not successfully complete two foundational courses to prove that they can succeed in the program.

Overall, attrition in the program happens for one of three reasons—the student fails to meet the admission requirement to move from a conditional accept to a full accept; the student underestimates the rigor and commitment required to succeed; or the student is just a knowledge seeker looking to hone their skills in a specific area, such as machine learning, and acquiring a degree is not an end goal. A small number of students get the "Ph.D. bug"; to date, more than 50 OMSCS students have entered CS Ph.D. programs, some at GT and others elsewhere.

TA support, counseling, and credit transfer. Finding adequate TA support for the courses is a challenge. We need about 400 TAs every semester. To meet this demand, we draw from four pools of students: OMSCS alumni, current OMSCS students, on-campus MS students, and on-campus Ph.D. students. In the spring of 2021, our TAs comprised 165 OMSCS alums (43%), 102 current OM-

^e See <http://omscs.gatech.edu/program-info/admission-criteria>.

SCS students (26%), 69 on-campus MS students (18%), and 50 on-campus Ph.D. students (13%). On-campus courses are typically staffed with a student-to-TA ratio of 30:1. Project-intensive advanced courses with moderate enrollments (less than 200) tend to have a ratio close to the on-campus courses. For courses with larger enrollments (for example, machine learning with an enrollment greater than 1,000), we use a 50:1 ratio. TAs are arranged hierarchically to ensure there is equitable division of labor and everyone does their fair share. The program also includes professional staff who help with the logistics of academic counseling and credit-transfer requests to ensure that faculty workload does not expand beyond the teaching responsibilities they sign up for.

Implications of the Program for the Future of Higher Education

While writing this article, the world was dealing with the COVID-19 pandemic. In the spring of 2020, academic institutions across the globe were scrambling to figure out how to provide academic programs to students without a physical, on-campus presence. Faculty who had been teaching OMSCS courses proved to be a great resource for peers within GT and beyond, helping to generate ideas about transitioning to the remote-learning format.

Most if not all OMSCS courses publish a schedule of the video lectures students are expected to watch and assimilate on a weekly basis as well as discuss amongst themselves in online forums. From our observations, much more peer learning occurs in the OMSCS format than in on-campus courses. Plus, many students bring a lot of “street knowledge” owing to their years of professional experience, which can be valuable even to the teaching team. Most if not all instructors offer weekly video hangouts for their courses, where students can get “face time” with instructors. Viewing lectures in advance lets students formulate insightful questions they can ask in hangouts.

Flipping the classroom for on-campus classes has not always met with as much success as one might expect due to the many demands on-campus students have on their time. The nature of the OMSCS program gives the flipped-classroom model a greater chance for

success; students know that a video hangout is their only chance for live interaction with faculty. Of course, due to the geo-distribution of students in the program, not all of them can attend hangouts. However, space-time issues can be mitigated by students who post questions ahead of time and access hangout recordings subsequently.

The OMSCS experience and the need to deal with remote instruction for even on-campus courses due to COVID-19 have given us food for thought on the pedagogy of higher education. Educators are discovering new, interesting ways to teach. Even when life returns to normal after COVID-19, there will be changes in how we teach students. One concrete example is a technique for facilitating peer learning.

To reduce student anxiety in taking timed tests online, one of this article’s authors invented a new method. Test questions are released to the entire class well ahead of time, allowing students to discuss the questions and solutions in messaging forums. All students must take a “timed closed everything test” at a time that suits their schedule (within a test-taking window spanning two days). The test is proctored using the same proctoring system we use for online students. From the student feedback we have received, this technique greatly reduced stress during the pandemic, and the intent is to continue using it even though GT returned to in-person lecturing for on-campus offerings as of the 2021 Fall semester.

Conclusion

OMSCS is a new way to provide a MOOC-based quality CS graduate program at scale and at a fraction of the cost of an on-campus education. It represents the fruits of a coordinated effort by the faculty and the CoC administration, plus creative partnerships with the industry. OMSCS, now reaching more than 11,000 students, may face technological challenges as it scales up to even larger class sizes. For example, student performance assessment cannot be entirely automated, streaming platforms may reach their scalability limits, and enlisting TAs as class size increases would become more challenging. GT CoC has been a pioneer in providing a high-quality, low-cost, MOOC-based graduate program in CS, yet there is much room

for growth as demand for a skilled CS workforce far exceeds the growth capacity of the OMSCS program. Since the initiation of OMSCS, more than 30 institutions have established over 70 similar, highly affordable MOOC-based online programs.⁵ We anticipate that many other institutions will follow suit.

Acknowledgments

We wish to thank Richard LeBlanc, Ayanna Howard, and Jim Foley for comments on an earlier draft of this manuscript. We also wish to thank Elijah Cameron, David White, David Joyner, and Reina Grundhoefer for the demographic data of the student population in the program and other statistics. The insightful comments of the anonymous reviewers and encouragement from Andrew Chien helped us to vastly improve the manuscript.

Postscript

As of the fall of 2022, 11,873 OMSCS students were currently enrolled, including 2,286 new admits. To date, the number of OMSCS graduates is 7,742; 2,213 graduated in the academic year 2021–22. ■

References

1. CRA-industry virtual roundtable on best practices on using the cloud for computing research. Computing Research Association (September 2021); <https://cra.org/crm/2021/09/cra-industry-virtual-roundtable-on-best-practices-on-using-the-cloud-for-computing-research-september-15/>.
2. Friedman, T.L. *The World Is Flat: A Brief History of the Twenty-First Century*. Farrar, Straus and Giroux (2005).
3. Galil, Z. OMSCS: The revolution will be digitized. *Commun. ACM* 63, 8 (2020), 27–29.
4. Güttel, C., Rizzardini, R.H., Chang, V., and Morales, M. Attrition in MOOC: Lessons learned from drop-out students. In *Learning Technology for Education in Cloud: MOOC and Big Data. Communications in Computer and Information Science* 446. Springer, Cham. (2014), L. Uden, J. Sinclair, Y.H. Tao, and D. Liberona, Eds.; https://doi.org/10.1007/978-3-319-10671-7_4.
5. Ledwon, H. and Ma, Rui. 70+ legit Master’s degrees you can now earn completely online.” *The Report by Class Central* (2021); <https://www.classcentral.com/report/mooc-based-masters-degree/>.
6. Proctortrack; <https://testing.verificient.com/login/>.

Umakishore Ramachandran (rama@gatech.edu) is a professor in the School of Computer Science, College of Computing, Georgia Institute of Technology, Atlanta, GA, USA.

Zvi Galil is the Emeritus Dean of Computing, Frederick G. Storey Chair of Computing, and Executive Advisor for Online Programs at the Georgia Institute of Technology, Atlanta, GA, USA.

© 2022 ACM 0001-0782/22/10 \$15.00



Watch the authors discuss this work in the exclusive *Communications* video. <https://cacm.acm.org/videos/overcoming-the-challenges>