Converting Upper-Division Undergraduate Computer Science Courses Online: Challenges, Student Performance, and Student Perceptions

Weiying Zhu
Department of Computer Sciences
Metropolitan State University of Denver
Denver, Colorado, USA
wzhu1@msudenver.edu

***Abstract*— This is a Research Full Paper. While considering offering online Computer Science (CS) courses at non-online programs in response to student demand or program need, how to address the challenges in online teaching and whether the student performance in an online course is compromised are two critical questions. In the literature, we could not find clear answers to these questions, especially for CS courses studied in this research in settings like ours. Converting three upper-division CS courses (operating systems, computer networks, and computer and network security) online in 2020-21 provided us an opportunity to explore the instructional design for online courses, to identify the challenges in online teaching, and to develop the strategies addressing challenges. We use t-tests to compare the student performances in online vs. face-to-face (F2F) sections of each course because our sample sizes fit in the most appropriate range (20 to 40) of t-tests. Box-and-whisker plots are also used to graphically compare student performances. Anonymous student surveys consisting of Likert-scale questions are used to investigate student perceptions in online courses. The analysis of performance and survey data shows no statistically significant performance difference between the online and F2F sections of each course taught by the same instructor as well as generally positive or very positive student feedbacks on teaching and learning in three online CS courses. Focusing on three neutral research questions, our study does not aim to comprehensively compare online and F2F modes or find which mode is better. Regardless what our data analysis indicates, we planned to report it. We hope that our experience and findings may provide useful and evidence-based information to other non-online computing or engineering programs, especially the ones like ours, when they need to consider whether to offer online courses.**

Keywords—online teaching, online learning, computer science, undergraduate courses, performance, student perception, survey

I. INTRODUCTION

Due to the advances in technology and the intention of making courses available to more students, more and more traditional non-online institutions are considering offering some courses online [3, 13, 15]. The benefits of offering online courses are twofold: 1) it provides flexibility that some students may need; and 2) it is a possible solution to some constraints in physical space or faculty. However, how to address the challenges in online Computer Science (CS) courses [8, 14] and whether the student performance in online CS courses is compromised are two critical questions to be answered before choosing to offer online CS courses, routinely or on demand. Our university is a large public Hispanic-Serving Institution. While there were some online courses at our institution before the pandemic, all CS courses used to be taught only face-to-face (F2F). In 2020-21, we taught CS courses online as a response to the pandemic. Unlike the sudden switch to online teaching in the middle of Spring 2020, this decision was made early enough for the university to build the infrastructure and for instructors to prepare for online teaching. This also gave us time to develop our research questions, choose research methods, and collect data in three online upper-division undergraduate CS courses.

In this research, we use case study to *qualitatively* research on the challenges in online teaching and the strategies to address them. We use t-tests and box-and-whisker plots to *quantitatively* compare student performance in online vs. F2F sections of each course. Our six sample sizes all fit in the most appropriate range of t-tests (20 to 40). For each course, a t-test is first performed between online students' cumulative GPAs and F2F students' to assure neither group has an unfair advantage. Next, we compare homework and exam scores in the online section and that in one or two F2F sections of each course. An anonymous survey was developed to *quantitatively* investigate student perceptions of online teaching and learning. Most of the survey questions are commonly used "agreement" Likert-scale questions for students to choose an answer ranging from "strongly disagree" to "strongly agree". Positive question statements do not make it less possible for students to truthfully reveal their opinions.

This paper presents our instructional design for online teaching, our approaches of supporting various online teaching and learning activities, the challenges we identified in assessing student learning, supporting learner interactions, supporting hands-on learning activities, and student time management in three online courses, and the strategies we developed to address the challenges. The performance and survey data collected in our courses are also analyzed in this paper. Our study has no intention to conclude which teaching mode is better: online or F2F. We believe they are different in many ways, and each may fit some students' individual learning styles or needs better. Our research questions are neutral. Regardless of what the data analysis indicates, we hope our findings may provide evidence-based information to non-online computing or engineering programs, especially the ones with similar settings to ours, when they need to consider whether to offer online courses.

II. Related Work

Studies on student performance in online computing courses have obtained different conclusions on whether the students in online sections perform worse than that in F2F sections. In [12], t-tests show that F2F students significantly outperformed online students in a Database course when there was no significant difference in students' cumulative GPAs. Comparing the exam score means of two hybrid and one online sections of a Software Process course [6], Fernandes *et al* concluded that online students performed worse than hybrid students. Campbell *et al*. observed a higher drop rate in their online CS1 than in the flipped section, given that there was a difference in student population between online and flipped sections [5]. Using an online sample and a F2F sample, each including 21 students, Bahamón et al found no significant difference in learning outcomes between course modalities in an Intro to C Programming course [3]. While converting a F2F Discrete Mathematics course online, Irani *et al* found no loss in student performance [8]. In [4], for an Intro to Digital Design course, Benson *et al* compared the performance of 23 students in an online semester and that of 65 students in a F2F semester, and found that online students performed as well as F2F students. Our t-test results reported in [20] regarding an Assembly Language and Computer Organization course indicate that there is no statistically significant difference between the student performance in an online section and that in F2F sections.

There have been some studies investigating online teaching and learning. Whitney et al explored the impact of captions on learning performance in an online Intro to Computers and Info Processing course [19]. Dividing 25 students into two groups to watch videos with or without captions, they found watching videos with captions resulted in better performance. In [11], using data collected on 12 female students and 37 male students, Little-Wiles et al studied gender differences in an online Ethical Decisions in Leadership course and found no significant difference between female and male students in final grades and LMS (learning management system) activities. In [2], based on 7 student responses in a hybrid section and 28 student responses in five online sections collected over three and a half years, Azemi discussed student ratings of teaching effectiveness in an introductory computer programming course. Analyzing survey data from civil engineering students after being forced to virtual learning due to quarantine, Singh et al found 15% of the students prefer online classes and 76% of them prefer synchronous online learning [16]. In [14], Rahimi et al discussed the challenges in online learning and teaching, e.g., greater difficulty in grasping concepts, delivering content, and managing group work.

III. Motivations, Courses, and Research Questions

At our university, we are considering whether to offer some CS courses online, routinely or on demand, for our non-online programs to provide the flexibility that some students need in pursuing higher education and to address the physical space limitation. However, we are concerned about the challenges in online teaching, student performance in online sections, and students' perceptions of online teaching and learning. Before the pandemic, we never taught any CS course online. Teaching and learning in online sections are very different from that in F2F sections. In the literature, there are different conclusions on whether students perform worse in online sections than in F2F sections. Moreover, the institutions, the student populations, or the courses reported in the literature are different from ours.

Therefore, this research studies three online upper-division undergraduate CS courses: Operating Systems (OS), Computer Networks (Net), and Computer and Network Security (Sec), taught in Year 2020-21 at our university. OS is a required core course for the B.S. in CS major program while Net and Sec are electives. All of them are electives for the B.S. in Computer Engineering (CpE) major program and the CS minor program. Below are three research questions of our study.

1. *What are the challenges of teaching online and supporting students' learning needs in three online upper-division CS courses, and the strategies to address these challenges?*
2. *Is there any student performance difference between online and face-to-face sections of the same course taught by the same instructor, and what is the difference if any?*
3. *What are student perceptions of their learning experiences and the online teaching tools and platforms, instructional materials, and hands-on activities in these online courses?*

Clearly, our study does not aim to comprehensively compare online teaching vs F2F teaching or to conclude which mode is better, even though student performance in one or two F2F sections of each course is used as a baseline to study the student performance in an online section of the same course.

IV. Research Methodology

This section discusses the methods used in our study, which are part of the theoretical foundation of this research.

A. Case Study to Identify Challenges and Strategies

To find the answer to our first research question, we chose to use case study, a *qualitative* research methodology that builds upon the analysis of single setting or occurrence [7] and is used to explore a particular instance in detail [17]. Unlike survey research, case study does not presume that difference instances can be thrown together to form a homogenous aggregate [7]. Although survey techniques may be used to collect data within it in a large case, case study collects and analyzes data about one instance while survey research studies many cases but a small amount of data about each one [10]. In contrast to experimental research that involves direct control of variables, case study explores the case in its naturally occurring context, instead of attempting to set up a clinic setting, because the natural context of a case often influences the characteristics of this case [10, 17].

The *instance* studied in this research is the teaching and the instructor's support to student learning needs in three online upper-division CS courses. Below is a description on the *natural context* of this instance. Aiming to provide a high-quality and accessible education with relatively low cost for students, our institution adopts less selective admission policies and has a high acceptance rate. Most of our students are first-generation college students. Our university is a commuter campus. The average age of our undergraduate students is about 25. Nearly 80% of students work while pursuing their education at our institution. Our CS program is ABET accredited. In our classes of undergraduate CS courses, there are a mix of students that

come with a wide range of previous knowledge and academic abilities. Compared to students in selective or highly selective CS programs at research or private institutions, our students are more diverse and generally face more academic and other challenges. To better work with a diverse student body, we keep the class size of almost all F2F CS courses between 20 and 34 and the class size of most online CS courses between 25 and 40. Our experience shows that such class sizes allow student-teacher interactions needed both during and outside class time.

B. T-Tests and Box-and-Whisker Plots to Compare Student Performances in Online v.s. F2F Sections

To compare the student performance when a course was taught online with the student performance when the same course was taught F2F, one or two sections of the *same CS course* taught by the *same instructor* during *most recent F2F semesters* are used as the *F2F* sample in this research. We followed two guidelines to decide the number of F2F sections used for a course: 1) to make the size of the *F2F sample* (consisting of all the students in the F2F section(s)) close to that of the *online sample* (consisting of all the students in the online section) and 2) to make the cumulative GPA mean of the F2F sample close to that of the online sample. The student learning objectives and course contents in F2F and online sections of each course are the same. No sections in Spring 2020 are used due to the unclear impact caused by the sudden switch to online.

The homework, exam, and term scores earned by every student in an online or F2F sample are used to compare student performance. A student's *homework* or *exam score* is the percentage of the total homework or exam points this student earned over a semester. Using the same weights for calculating a student's final term grade, a student's *earned term score* is equal to 50% of this student's homework score plus 50% of this student's exam score. Any extra bonus point given to students for grade promotion or extra work is not included in any of a student's three scores.

Based on the size of every data set (i.e., cumulative GPAs, homework scores, or exam scores of the students in an online or F2F sample), the first *quantitative* analysis technique we chose to use is t-test [9], a commonly used statistical hypothesis test. The appropriate sample size for performing a t-test is between 20 and 40. If the sample size is much higher than this range, a z-test should be used. This research uses two-sample location tests of null hypotheses, each of which states that there is no effective difference between the mean of an online dataset and the mean of a F2F dataset (i.e., any measured difference is only due to chance). The t-tests performed in this research are *two-tailed* tests because each is to test for the possibility of the relationship in both directions: no matter whether one mean is effectively greater or less than the other, two means are effectively different. The online sample and F2F sample of each CS course in this study are *independent unpaired* samples. Determined by the data sets, every t-test performed in this work is for two samples with different sizes but with similar variances. Every t-test calculates a *p-value*, i.e., a probability value to support or reject the null hypothesis. When the calculated p-value is below the chosen significance level (i.e., the α value that is usually 0.10, 0.05, or 0.01), the null hypothesis is rejected. Otherwise, the null hypothesis is supported. The *higher* the p-value is, the *stronger* the evidence *supporting* the null hypothesis is.

TABLE I. AGREEMENT LIKERT-SCALE QUESTIONS ON THE USE OF TOOLS, INSTRUCTIONAL MATERIALS, AND ACTIVITIES

	Questions in a concise version
Q4	Despite a possible learning curve, it's easy to find *weekly* learning materials in Canvas.
Q5	Despite a possible learning curve, it's easy to find *general* learning materials in Canvas.
Q6	Despite a possible learning curve, it's not difficult to submit work in Canvas.
Q7	Despite a possible learning curve, it's not difficult to use virtual Linux servers.
Q8	Posting lecture videos before a week starts makes it more *flexible* to take lectures.
Q9	Pausing/replaying lecture videos allows self-paced learning and is beneficial to learning.
Q10	Pausing/replaying videos allows following the instructor's demos to repeat hands-on activities.
Q11	Having access to lecture videos outside class time is helpful for *working on homework assignments*.
Q12	Having access to lecture videos is helpful *when not being able to attend class*.
Q13	The use of virtual Linux servers is helpful for improving the programming skills.
Q14	The use of virtual Linux servers allows debugging code in the same environment that the instructor uses for testing and grading.
Q15	Overall, *lecture videos* are useful for learning.
Q16	Overall, *lecture notes* are useful for learning.
Q17	Overall, *PPT slides* are useful for learning.
Q18	Overall, *example programs* are useful for learning.
Q19	*Hands-on activities and labs* are helpful for achieving learning objectives.
Q20	*Programming assignments* are helpful for achieving learning objectives.
Q21	If it were F2F, you would still want *lecture videos* to be provided.
Q22	If it were F2F, you would still want *lecture notes* to be provided.
Q23	If it were F2F, you would still want *PPT slides and example programs* to be provided.

A t-test was first performed between the cumulative GPAs of the online sample and that of the F2F sample for each CS course to compare students' background demographics in terms of their cumulative GPAs. Next, t-tests were performed between students' homework or exam scores in the online section and that in the F2F section(s) for each course to compare student performance in online vs. F2F teaching modes.

The other *quantitative* analysis technique we chose for performance comparison is the box-and-whisker plot [18], a descriptive statistical tool that graphically displays variation in a data set. A box-and-whisker plot is a standardized way of displaying a data set via its maximum, minimum, median, upper quartile, and lower quartile. A box-and-whisker plot may include outliers. If it does, two whiskers are drawn, one for the largest data and the other for the smallest data excluding outliers.

For each CS course studied in this research, box-and-whisker plots are drawn to graphically describe students' earned term scores in the online and F2F samples and to visually compare the degree of dispersion and skewness of the scores in the online sample and that of the scores in the F2F sample.

C. Anonymous Surveys to Investigate Student Perceptions

A *quantitative* student survey was designed to get students' perceptions of the learning experience and their opinions on the

instructional materials and learning activities in three online CS courses studied in this research. The first three questions (Q1 through Q3) are about the negative impact caused by the pandemic and related situations on students' availability during the class time or on their time commitment, or about the technical difficulty of taking lessons online during class time. Followed are twenty Likert scale questions on the use of tools, instructional materials, and hands-on learning activities. Table I shows a concise version of Q4 through Q23. Note that these are Agreement Likert scale questions, the most common type of Likert scale questions and responses, where students were given five choices ranging from "strongly disagree" to "strongly agree". Also, students took surveys anonymously. Therefore, having those statements positive does not make it less possible for students to truthfully reveal their opinions.

V. Instructional Design and Challenges in Three Online Upper-Division Undergraduate CS Courses

The data collection in the case study to identify the challenges in three online upper-division undergraduate CS courses and the strategies to address them mainly relies on instructors' observation and documentation.

A. Instructional Design for Online Teaching

The instructional design in our online courses follows the recommendations in the Quality Matters Rubric [1] regarding eight course components. The discussion in this section focuses on what we specially designed for online teaching and learning.

a) Course Technology. The tools we specially adopted to support online course delivery and to manage online learning are Canvas[1], MS Teams[2], and MS Stream[3]. We also continued using virtual Linux servers to support hands-on activities, using GlobalProtect [4] to support virtual private network (VPN) connections from outside to the campus network, and using emails for student-instructor communication.

b) Learning Objectives. The learning objectives of the online version and that of the F2F version of each course are the same.

c) Overview and Instruction. In each online CS course, a 50-minute virtual meeting was held in MS Teams on the first day for overview and instruction. This meeting focused on specific info for online learning, such as where to find instructional materials in Canvas, the schedule of virtual meetings for Labs and Exam Q&As and where to find meeting links, how and where to reach the instructor inside and outside the class time and office hours, and the info on online exams. All course policies and information were posted as pages in the "Course Information" module, the first one in Canvas. In this module, a Discussion page was created for student self-introduction.

d) Instructional Materials. To teach online, the newly developed instructional materials are the lecture videos that the instructor pre-recorded and uploaded to MS Stream for most class periods. Even though it was very time consuming, the instructor did it for better quality and availability. PPT slides and the resources and examples for C programming (in OS) or Java programming (in Net and Sec) were posted as pages in the second and third modules in Canvas. Underneath the top three modules are "Weekly Learning Activities and Materials" modules, one for each week of a semester. In every weekly module, a page was created for each class day to publish either the links to pre-recorded lecture videos together with the lecture notes or the link to a virtual meeting in MS Teams scheduled for the class time. Every weekly module was made available to students in Canvas before a week started.

e) Learning Activities and Learner Interaction. OS, Net, and Sec each includes a variety of hands-on learning activities such as programming assignments, projects, labs, and hands-on practice. Here are some examples: programming in C using the POSIX library (in OS), client and server programming for application-layer services using UDP or TCP sockets (in Net), client and server programming using Java Secure Socket Extension (in Sec), cryptography programming (in Sec), file and directory operations on remote Linux servers via an SSH client (in OS, Net, and Sec), secure file transfer operations between a local computer and a remote Linux server via an SFTP client (in OS, Net, and Sec), making queries to retrieve DNS resource records and exploring various types of name servers in the DNS hierarchy and the hosts in selected domains (in Net), and exploring file access control for UNIX and UNIX-alike systems and relevant commands on a Linux server (in Sec). Additionally, paper-based assignments are used for students to practice on concepts, methods, algorithms, or procedures and to practice on applying them to solve problems. The interactions between the instructor and the learners in these upper-division CS courses are needed during the class time, during the instructor's office hours, and outside the class time and office hours.

Supporting the learning activities and learner interactions in online courses are more challenging than that in F2F courses. We eventually were able to support all of them remotely online using various approaches, most of which are different from how we supported them in F2F CS courses.

f) Assessment and Measurement. Besides being part of the learning activities, homework and projects were also used to assess student learning. A project or homework assignment was created as an "Assignment" in Canvas, with a submission type of "Online" and an online entry option of "File Uploads", for students to submit the source code of their programs or an electronic copy of their paper-based homework. Additionally, students were required to upload and test their programs under their home directories on the class virtual Linux servers.

Exams were the other critical component in assessing and measuring student learning outcomes. 50% of a student's final term grade came from his or her homework and project average score and the other 50% came from his or her exam average score. All exams were made into online tests, i.e., "Quizzes" in Canvas. The length of an online test was made the same as that of its corresponding F2F exam. The question types used in online tests include "Essay Question", "Fill in Multiple Blanks", "Multiple Choice", and/or "True/False".

g) Learner Support, Accessibility and Usability. In these online CS courses, the instructor, our department, and our

[1] https://www.instructure.com/canvas
[2] https://www.microsoft.com/en-us/microsoft-teams/group-chat-software
[3] https://www.microsoft.com/en-us/microsoft-365/microsoft-stream
[4] https://www.paloaltonetworks.com/products/globalprotect

institution continued providing the same support to learners such as tutoring services and IT services and the same support to accessibility and usability as what was provided in F2F courses, except that all such services and interactions were provided via one-to-one virtual meetings in MS Teams.

B. Challenges and Our Strategies to Address Them

Based on instructors' observations and students' feedbacks, below is a discussion on the challenges that we observed in these three online CS courses and the strategies that we developed to address or mitigate the challenges.

a) Assessing student learning via online tests is the greatest challenge that we experienced in these three online CS courses. This challenge is twofold: 1) it was more difficult for the instructor to enforce academic integrity and to make online tests, especially the test questions that need to include diagrams in the solutions; and 2) *some* students reported that they were not used to online tests and it took extra time for them to read through all instructions specifically written for online questions and/or to go to MS Teams to send questions via CHAT messages to the instructor for clarification during an online test. Even though there exist tools like ProctorU, it was infeasible to effectively proctor online tests. Not all students had webcams and some students reported privacy or security concerns about installing software like ProctorU on their computers. Moreover, even if all students' testing computers had webcam and ProctorU, the easy access to a second computing device would make remote proctoring ineffective. It took much more effort and time to develop an online test than developing a paper-based test. Alternative text format needed to be developed for presenting any solution that involves diagrams and needed to be clearly described in the problem statement for students who cannot take a clear photo of or scan their solutions written on paper. Multiple versions of every problem needed to be developed to promote academic integrity.

We developed several strategies to make it harder to violate academic integrity or to make it easier to catch such violations in online tests. First, most of the points in an online test were for problems that require students to analyze and solve using the specific info given in the problem statement. Students were required to include sufficient details in their solutions. Less than 10% of the points were for problems on concepts and basic facts. Secondly, a "Question Bank" consisting of multiple questions was created in Canvas for every problem in an online test and one of those questions was randomly picked for a student. This significantly reduced the chance of any two students getting the same set of questions. Thirdly, the questions in an online test were shown one at a time and a question was locked when a student moved to the next one. Fourthly, all students in class were required to take an online test simultaneously. Fifthly, students were reminded before every online test about details of academic integrity and consequences of violating it. Lastly, when anything suspicious was observed in the submitted work, the "View Log" function under "Moderate This Quiz" in Canvas was used to check student actions during an online test.

b) Supporting learner interaction is also a major challenge in online courses. Unlike straightforward learner interactions during the class time in the classroom or during the instructor's office hours in the instructor's office in a F2F course, the instructor needed to make extra effort to support and encourage students to interact with the instructor in an online course.

Several approaches were used together to provide as many opportunities for learner interactions as we can. For each online CS course, the instructor created a virtual classroom (team) in MS Teams and posted real-time replies to student questions that were posted during class time. The instructor reserved several class periods for questions and answers (Q&As) regarding in-class labs and exams. Instead of prerecording lecture videos for these class periods, the instructor held live virtual meetings in MS Teams for students to ask questions. The instructor hosted virtual online office hours in MS Teams, during which students sent a CHAT message to the instructor and then the instructor called students back via a CHAT video or audio call that allows screen sharing. Outside class time and office hours, the instructor encouraged students to send their questions via emails and promptly replied to student emails. Besides, providing students timely feedback to their homework is also an important piece of learner interactions. The instructor electronically graded homework submissions in a timely fashion and posted graded work with details of grading and corrections (for non-programming homework) or with details of testing results (for programming homework) in Canvas.

c) Supporting hands-on learning activities is another major challenge in online courses. In a F2F course, we count on in-class labs and the instructor's office hours to help students on hands-on activities. In classrooms or instructors' offices, students show problems or error messages on their computers to the instructor and the instructor helps students troubleshoot. Such individualized help in our F2F courses was shown to be essential to student learning because different students encountered different difficulties or were confused in different ways. Also, all hands-on learning activities in these three online CS courses need to be done on the virtual Linux servers inside our institutional private cloud. To connect to those virtual Linux servers from computers outside the campus network, a connection to the institutional VPN needs to be established first.

To support hands-on learning activities in these three online CS courses, we recorded lab or lecture videos to demonstrate the commands, procedure, or method relevant to every hands-on activity, uploaded videos to MS Stream, posted the video links in Canvas, and hosted pre-scheduled or on-demand virtual meetings in MS Teams during class time or CHAT video/audio calls during office hours, all of which allowed students to share their computer screens, to help students troubleshoot. We also continued requesting IT services to set up virtual Linux servers for every online class and requesting the creation of a GlobalProtect account for every student in class.

d) Student time management is a challenge faced by some students in our online CS courses. Without the need of going to campus for classes and office hours, some students reported that it was difficult for them to keep up with schoolwork and they tended to spend the time that were supposed to be spent for classes, learning activities, and homework on something else. Unfortunately, the flexibility that was provided in online courses made it easier for some students to postpone their learning by thinking they would do it later, which usually did not work and caused too much work accumulated to be completed.

TABLE II. TEACHING AND LEARNING IN OUR ONLINE VS. F2F OS, NET, AND SEC COURSES: DIFFERENCES AND SIMILARITIES

Teaching/learning materials or activities	Online courses	Face-to-Face courses
Lecturing	The instructor pre-records lecture videos, uploads them to MS Stream, and posts the lecture video links together with lecture notes in Canvas before every class day. Students are recommended to watch lecture videos during the class time and have access to lecture videos throughout the rest of the semester.	The instructor gives lectures in the classroom during class time. Outside class time, students have no access to lectures even though lecture notes are posted in Blackboard, the LMS we used to use before switching to Canvas.
In-class labs (for hands-on activities on virtual Linux servers)	Students first connects their home computers to the institutional VPN via a GlobalProtect client and then connect to the virtual Linux servers set up by the IT services for the class.	The instructor reserves an on-campus computing lab. Students use lab computers to connect to the virtual Linux servers set up for the class.
In-class project presentations	The instructor creates virtual meetings in MS Teams for students to present and demonstrate their work via screen sharing.	Students present and demonstrate their work in the classroom.
Instructional materials	All, including lecture videos, PPT slides, lecturing SmartBoard/OneNote notes, source code of example programs, lab manuals, are posted in Canvas and MS Stream.	All, including PPT slides, in-class SmartBoard notes, source code of example programs, lab manuals, are posted in Blackboard.
Learner interactions	Students post questions in the virtual classroom in MS Teams during class time for the instructor to post real-time replies. The instructor uses CHAT voice/audio calls in MS Teams to hold virtual office hours. Emails are used for learner interactions outside class time/office hours. Announcements are made via "Announcement" in Canvas, posts in MS Teams, and emails.	Students ask questions and the instructor answers questions during class time in the classroom and during office hours in the instructor's office. The instructor makes announcements both in class and via emails.
Homework	The instructor collects the submissions of non-programming work in Canvas and collects the submissions of student programs both in Canvas and on virtual Linux servers.	The instructor collects the submissions of non-programming work on paper during class time and collects the submissions of student programs both in Blackboard and on virtual Linux servers.
Exams	The instructor develops and gives closed-book & closed-notes online tests in Canvas. However, online tests cannot be proctored.	The instructor gives and proctors closed-book & closed-notes paper-based tests in the classroom.

It surely is students who had to take actions to address this challenge. But we made several efforts to hopefully help students with better time management. We organized the lecture videos, lecture notes, and lab manuals by class days (via one page in Canvas per class day) and then organized them together with homework or project pages into weekly learning materials & activities modules in Canvas, one module per week. We published the due date, cutoff date deadline, and total points of every homework or project assignment and the dates of all exams at the beginning of a semester. The policy of not accepting submissions after cutoff deadlines had been made clear to students via several ways ever since the first day of class.

Table II summarizes the approaches to support teaching and learning in our online OS, Net, and Sec courses, and shows how they are different from or similar to that in F2F courses.

VI. STUDENT PERFORMANCE ANALYSIS AND COMPARISON

The F2F samples used to compare student performances are OS in Fall 2019 (size = 30), Net in Spring 2018 and Summer 2017 (size = 36), and Sec in Fall 2019 (size = 24). The online samples are OS in Fall 2020 (size = 37), Net in Spring 2021 (size = 31), and Sec in Fall 2020 (size = 23). All the sample sizes fit in the most appropriate range of t-tests (i.e., 20 to 40).

A. Student Background Demographics

Assured by the prerequisites, these courses are taken by seniors and juniors. To provide more information on the student population in each class, Table III shows student background demographics in every online or F2F sample in terms of majors (MTH: Mathematics), genders, races/ethnicities (Wht: white, Hsp: Hispanic, Asn: Asian American, Afr: African American, Oth: Other – mostly unknown/missing), and ages. It is shown that most students are majored in CS, male, and white (although there are more Hispanic students in more recent semesters). The average student ages are between 25 and 30.

TABLE III. BACKGROUND DEMOGRAPHICS: MAJOR, GENDER, RACE/ETHNICITY, AND AGE

	Online					F2F				
Major	CS	CpE	MTH	Other		CS	CpE	MTH	Other	
OS	92%	0%	3%	5%		100%	0%	0%	0%	
Net	87%	6.5%	0%	6.5%		89%	0%	8%	3%	
Sec	91%	4.5%	4.5%	0%		92%	0%	8%	0%	
Gender	Male		Female			Male		Female		
OS	86.5%		13.5%			83.3%		16.7%		
Net	77.4%		22.6%			80.6%		19.4%		
Sec	95.7%		4.3%			95.8%		4.2%		
Race/Eth	Wht	Hsp	Asn	Afr	Oth	Wht	Hsp	Asn	Afr	Oth
OS	46%	27%	14%	5%	8%	57%	13%	13%	7%	10%
Net	45%	32%	3%	7%	13%	67%	14%	14%	0%	5%
Sec	56%	26%	9%	9%	0%	58%	13%	4%	8%	17%
Average Age (Yrs)	**OS**	**Net**	**Sec**			**OS**	**Net**	**Sec**		
	25.3	28.0	27.6			26.7	27.6	29.4		

Even though students' background demographics in terms of majors, genders, races/ethnicities, and ages in the online and F2F samples of each course are not identical with each other, it does not mean any sample has an unfair advantage. Firstly, each CS course has well-defined prerequisites that were enforced in all sections. Secondly, the regression analysis in [12] shows that

cumulative GPA, not any other demographic, is the single most important variable when term grade is the dependent variable.

Table IV describes the background demographics of online and F2F samples through the means and standard deviations of students' cumulative GPAs. Clearly, the two samples of each course have cumulative GPA means close to or even identical with each other. The biggest difference is in Sec, where the F2F sample has a mean 2.5% higher than that of the online sample. A t-test (α=0.05) was performed between the two samples of each course and the t-test results are shown in Table V. All p-values are much higher than 0.05. This indicates that there is no statistically significant difference between the cumulative GPA mean of the online sample and that of the F2F sample for each course. Therefore, neither sample of each course has an unfair advantage in terms of student cumulative GPAs.

TABLE IV. BACKGROUND DEMOGRAPHICS: CUMULATIVE GPAS

	Online Mean	Online Std-Dev	**F2F Mean**	F2F Std-Dev
OS	**3.10**	0.474	**3.15**	0.526
Net	**3.24**	0.574	**3.24**	0.688
Sec	**3.17**	0.635	**3.25**	0.401

TABLE V. RESULTS OF T-TESTS ON CUMULATIVE GPAS (α = 0.05)

	t-Value	T Critical Value (two-tail)	***p*-Value**	**Significant?**
OS	-0.3533	±1.997	**0.7250**	**No**
Net	0.0104	±1.997	**0.9917**	**No**
Sec	-0.5487	±2.014	**0.5859**	**No**

B. Student Performance Comparison via T-Tests

Tables VI and VIII compare the homework and exam score means and standard deviations of the online sample and that of the F2F sample, respectively. We can see that the homework or exam score means of two samples for each course are very close to each other while some online means are higher than their corresponding F2F means and some online means are lower. The biggest difference (only 2.4%) is between the homework score mean of online students and that of F2F students in OS.

TABLE VI. HOMEWORK SCORES: MEANS & STANDARD DEVIATIONS

	Online Mean	Online Std-Dev	**F2F Mean**	F2F Std-Dev
OS	**75.0%**	18.0%	**72.6%**	18.4%
Net	**74.2%**	24.5%	**74.7%**	25.6%
Sec	**72.6%**	20.8%	**73.4%**	18.3%

TABLE VII. RESULTS OF T-TESTS ON HOMEWORK SCORES (α = 0.05)

	t-Value	T Critical Value (two-tail)	***p*-Value**	**Significant?**
OS	0.5532	±1.997	**0.5821**	**No**
Net	-0.0900	±1.997	**0.9285**	**No**
Sec	-0.1414	±2.014	**0.8882**	**No**

TABLE VIII. EXAM SCORES: MEANS & STANDARD DEVIATIONS

	Online Mean	Online Std-Dev	**F2F Mean**	F2F Std-Dev
OS	**66.6%**	22.5%	**65.3%**	21.2%
Net	**65.6%**	21.4%	**67.6%**	26.4%
Sec	**67.2%**	26.3%	**68.7%**	13.3%

TABLE IX. RESULTS OF T-TESTS ON EXAM SCORES (α = 0.05)

	t-Value	T Critical Value (two-tail)	***p*-Value**	**Significant?**
OS	0.2454	±1.997	**0.8069**	**No**
Net	-0.3410	±1.997	**0.7342**	**No**
Sec	-0.2574	±2.014	**0.7980**	**No**

However, simply comparing the corresponding means of two samples or having the score mean of one sample greater than that of the other sample (especially when two means are close) cannot indicate that the students in one sample performed better than the students in the other. Therefore, t-tests were performed to compare the homework or exam scores of the online sample and that of the F2F sample for each course. T-tests results (α = 0.05) are shown in Tables VII and IX. A positive t-value shows that the online mean is greater than the F2F mean, and a negative t-value shows that the online mean is smaller. We can see that all six p-values (0.5821 or higher) are much greater than 0.05. Every t-value is within the critical range bounded by a pair of opposite *t* critical values and is close to or very close to the center point (0) of the critical range. Altogether, it is shown that there is no statistically significant difference between the performance of online students and the performance of F2F students for each course in terms of homework and exam scores.

C. Student Performance in Box-and-Whisker Plots

Figure 1 shows the box-and-whisker plot of earned term scores, defined in Section IV.B, in every online or F2F sample. For OS, the upper-quartile, median, mean, and lower-quartile of the earned term scores in the online sample are all slightly higher or higher than their corresponding values of the earned term scores in the F2F sample, respectively. For Net, the median and mean of the earned term scores in the online sample are very close to that in the F2F sample, respectively, while the online sample has smaller interquartile range and smaller range than the F2F sample. For Sec, the online and F2F samples have almost the same median and mean with each other. The online sample's interquartile range and range are both greater than that of the F2F sample. For each course, the online sample include one or a few more outliers than the F2F sample. This further shows that, in each course, the performance of online students is statistically close to or not worse than that of F2F students.

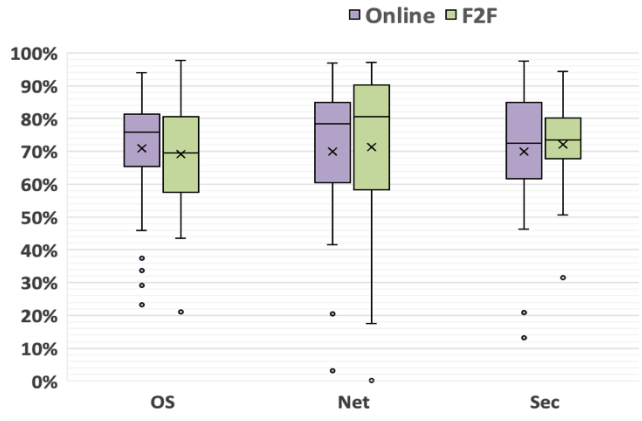

Fig. 1. Box-and-whisker plots of earned term scores.

VII. ANALYSIS OF SURVEY DATA

Near the end of a semester, an *anonymous* survey in Canvas was given to students in each online course. We received the answers from 30 students in OS (Fall 2020), 28 students in Net (Spring 2021), and 20 students in Sec (Fall 2020).

Figure 2 shows the mean of students' ratings of the negative impact caused by the ongoing pandemic or related situations on their availability during the class time (Q1) or on their time commitment to this course (Q2) or that of the technical difficulty to take lessons online at class time (Q3). Apparently, among these challenges, the greatest one facing students is the negative impact on their time commitment to a course, which by average was rated somewhere between "low" and "medium". Also, pre-recording lecture videos and making them available to students before class time somehow mitigated the effects of the negative impact on some students' availability during class time and the effects of technical difficulty for some students to take lessons online at class time.

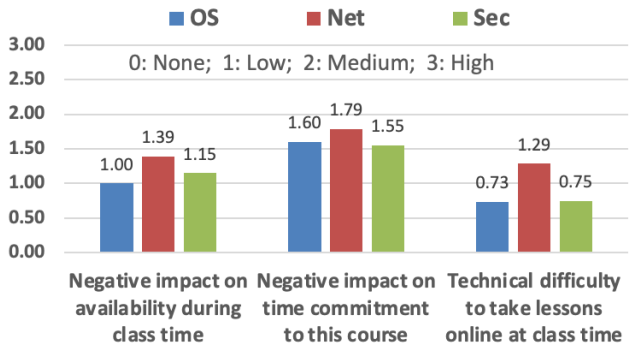

Fig. 2. Means of ratings of negative impacts and technical difficulty.

Figure 3 plots the mean of the Liker scale ratings given by students in each online course to each of questions Q4 through Q23, listed in Table I. We can see that all means are between 4 (Agree) and 5 (Strongly Agree). This shows a generally positive or very positive feedback from students on their experiences of using the tools and regarding their opinions on instructional materials or hands-on learning activities in these online CS courses.

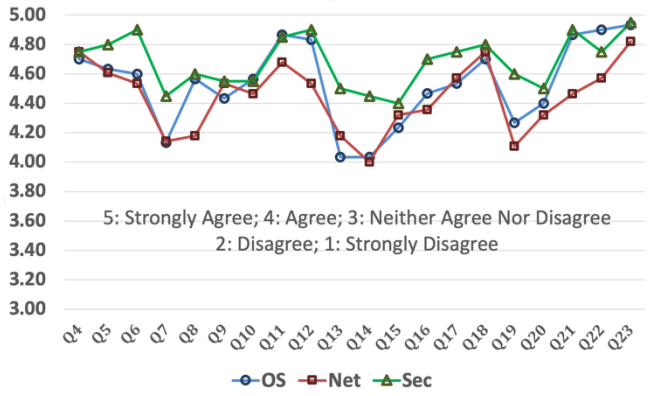

Fig. 3. Means of Likert scale ratings for Q4 through Q23.

Particularly, the questions receiving rating means that are 4.50 or above in all three online courses are Q4, Q5, Q6, Q11, Q12, Q17, Q18, Q22, and Q23. This indicates that, in all three online courses, students' feedbacks are generally very positive on their experiences of using Canvas, using lecturing videos for working on homework assignments and for catching up missed classes, using PPT slides, and using example programs. Meanwhile, we can see that student ratings to these questions in Sec are generally slightly higher than that in OS and Net. This finding is interesting but not surprising. Even though these three courses were all taught online and the same tools and platforms were used to support online teaching and learning, their course contents, learning objectives, programming and paper-based homework assignments, and hands-on activities are all different from each other. There is a 5-week term project on cryptography in Sec, which includes student presentations, while there isn't any term project in OS and Net. Furthermore, security has been a hot area. Due to the various security incidents reported in the news, students can easily see the needs of pursuing the security of computing systems and networks. This might affect how much time students spent on their learning in a course and, thus, might also affect their perceptions to their learning experiences in this course.

VIII. Conclusions and Future Work

In this paper, we present three research questions that were motivated by the program needs at our university, the methods we chose to research on our questions, the instructional design and our approaches to support various learning and teaching activities in three upper-division undergraduate CS courses, the challenges identified for online teaching and learning, our strategies to address or mitigate challenges, the quantitative analysis of the performance and survey data collected in our courses, and the analysis results. It is a little surprising but encouraging to find no loss in student performance in three online upper-division undergraduate CS courses. The data analysis of anonymous surveys indicates positive student feedback on the teaching and learning in these online courses. Our findings give us the confidence to include offering online upper-division CS courses as an option, if needed, in the future.

This paper also describes the characteristics of our courses and our university as well as the background demographics of the students in three upper-division CS courses. By providing such information, we hope that non-online programs at other institutions, especially the ones like ours, may make use of our approaches, strategies, or findings when they need to consider whether to offer or need to offer online sections of computing or engineering courses, routinely or on demand.

In future online and F2F sections of these three upper-division undergraduate CS courses or in other undergraduate CS courses taught by other instructors, we plan to collect more assessment and survey data to replicate this research on a larger scale so that we can further generalize our findings. Also, utilizing the lecture videos that we spent a lot of time making and specifically customizing for the learning objectives and course contents of these three CS courses, we plan to explore the flipped-classroom approach in these courses, study its impact to student learning, and survey student perceptions.

References

[1] Standards from the Quality Matters Higher Education Rubric. Available: https://www.qualitymatters.org/qa-resources/rubric-standards/higher-ed-rubric

[2] A. Azemi, "Examining students' concerns related to online delivery of an introductory computer programming course," In Proceedings of 2016 IEEE Frontiers in Education Conference (FIE), 2016, pp. 1-4.

[3] J. C. Bahamón and A. Rorrer, "Improving student learning outcomes in

online courses: an investigation into the effects of multiple teaching modalities," In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20), pp. 1179–1185, March 2020.

[4] B. Benson and B. Mealy, "Experience with teaching digital design online," In Proceedings of the 2014 IEEE Frontiers in Education Conference (FIE), October 2014.

[5] J. Campbell, D. Horton, and M. Craig, "Factors for success in online CS1," In Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '16), pp. 320–325, July 2016.

[6] E. Fernandes, J. Oliveira, and E. Figueiredo, "Investigating how features of online learning support software process education," In Proceedings of 2016 IEEE Frontiers in Education Conference (FIE), 2016, pp. 1-8.

[7] D. Hamilton, "Some contrasting assumptions about case study research and survey analysis," In H. Simons (Ed.), Towards a scient of the singular: Essarys about case study in educational research and evaluation, Norwich, UK: Centre for Applied Research in Education, University of East Anglia, pp. 78-92, 1980.

[8] S. Irani and K. Denaro, "Incorporating active learning strategies and instructor presence into an online discrete mathematics class," In Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20), pp. 1186–1192, March 2020.

[9] D. G. Kleinbaum, L. L. Kupper, A. Nizam, and E. S. Rosenberg, "Applied regression analysis and other multivariable methods," 5th. ed., Cengage Learning, Boston, MA, 2013.

[10] M. S. Lewis-Beck, A. Bryman, and T. F. Liao, "Case study," In M. Lewis-Beck (Ed.), The SAGE Encyclopedia of Social Science Research Methods, SAGE Publications, pp. 93-94, 2004.

[11] J. Little-Wiles, E. Fernandez, and P. Fox, "Understanding gender differences in online learning," In Proceedings of 2014 IEEE Frontiers in Education Conference (FIE), 2014, pp. 1-4.

[12] M. McDonald, B. Dorn, and G. McDonald, "A statistical analysis of student performance in online computer science courses," In Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '04), pp. 71–74, March 2004.

[13] I. Naimi-Akbar, L. Barman, and M. Weurlander, "Engineering teachers' approaches to teaching and learning online," In Proceedings of the 2020 IEEE Frontiers in Education Conference (FIE), 2020, pp. 1-5.

[14] N. Rahimi and N. L. Martin, "Challenges and Strategies for Online Teaching in information technology and other computing programs," In Proceedings of the 21st Annual Conference on Information Technology Education (SIGITE '20), pp. 218–222, March 2020.

[15] P. Raviolo, "Online higher education teaching practices," In Proceedings of the 10th International Conference on E-Education, E-Business, E-Management and E-Learning (IC4E '19), pp. 79-84, January 2019.

[16] M. Singh, E. Clay, Y. Salem, and W. Cheng, "Determining the students' preferable learning mode for both traditional classrooms teaching under normal situations and forced virtual teaching in quarantine period," In Proceedings of 2021 IEEE Frontiers in Education Conference (FIE), 2021, pp. 1-8.

[17] K. S. Taber, "Methodological issues in science education research: a perspective from the philosophy of science," In M. R. Mathews (Ed.), International Handbook of Research in History, Philosophy and Science Teaching, Vol 3, Dordrecht: Springer Netherlands, pp. 1839-1893, 2014.

[18] J. W. Turkey, Exploratory Data Analysis, 1st. ed., Addison-Wesley, Boston, MA, 1977.

[19] M. Whitney and B. Dallas, "Captioning online course videos: an investigation into knowledge retention and student perception," In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), pp. 511–517, March 2019.

[20] W. Zhu, "Challenges and experiences of conversting an assembly language and computer organization course into an online course," In Proceedings of American Society for Engineering Education (ASEE) 2022 Annual Conference, pp. 1–16, June 2022.