COMPUTER SCIENCE

Office: Ritchie School of Engineering & Computer Science
Mail Code: 2155 E Wesley Avenue, Room 379, Denver, CO 80208
Phone: 303-871-2458
Email: info@cs.du.edu
Website: Computer Science (http://www.du.edu/rsecs/departments/cs/)

Master's and Doctoral degrees

Why study Computer Science at the University of Denver?

DU's Department of Computer Science (CS) is creating the future of technology by providing a graduate education that emphasizes both multi-disciplinary and cross-disciplinary knowledge. The distinguished faculty are creating multi-disciplinary education programs that cover both depth and breadth, and research programs that incorporate and account for technological trends in research and development, along with industry. Graduate students join the faculty in conducting cutting-edge basic and applied research in emerging disciplines developing novel and unique solutions to old and new problems and opportunities. Additionally, students are not required to have a computer science background as we offer foundational courses to prepare students for the road ahead.

All laboratories in the Department contain state-of-the-art equipment and software to support research in algorithms, artificial intelligence, computational geometry, humane games, networks, programming languages, robotics, human-robot interaction, security and privacy, and software engineering, among other research areas. Small classes support our multi-disciplinary and real-time focus by providing close contact between students and faculty, which allows us to meet students' individual career goals.

Denver is a first-rate location for business, government and laboratory partnerships, and technology employment. The Colorado Front Range is consistently rated as one of the top high-tech areas in the country, and DU is located just minutes from the Denver Technological Center, the site of many top technology companies. The Department of Computer Science is committed to active collaboration with these industry leaders. As a result, our students graduate with relevant research experience and a network of employment contacts in the technology sector that is second to none!

Degree Programs

We offer the following graduate computer science degrees:

- MS in Computer Science
- MS in Cybersecurity
- MS in Data Science
- PhD in Computer Science

Doctor of Philosophy in Computer Science

The PhD in Computer Science is, above all, a research degree. When the faculty award a PhD, they certify that the student has a broad foundation and awareness of core concepts in computer science, has advanced the field by performing significant original research and has reported that work in a scholarly fashion. Through this degree, students will be qualified to hold academic positions at universities across the world, to lead research groups in government or industry research laboratories, to be creators of new ideas and approaches in computer science through research, and to be leaders in technology. Our faculty are strong in research and particularly noted in software engineering, robotics, cybersecurity and privacy, and humane games.

Master of Science in Computer Science

The MS in Computer Science prepares students for advancement in academic or industrial careers. The program is designed to provide students with a breadth of advanced knowledge in computer science, while permitting them to achieve depth in areas of current interest within the computing field through electives, as well as the emerging technologies that will have great importance in the future. Students of this degree will be eminently qualified for more advanced careers using computer science given the ubiquitous nature of computing today in every single discipline. Students may also pursue a thesis option, enabling them to begin to perform research and be prepared for a PhD in Computer Science or related discipline.

Master of Science in Cybersecurity

The MS in Cybersecurity prepares students for a change of careers into the cybersecurity field, or for advancement of a current career. Malicious actors, from small-time hackers to nation states, are an ever present threat to users on the internet today. Cybersecurity expertise is invaluable in any field, as nearly all important personal information is stored online and business is increasingly done online. The cybersecurity MS program is designed to provide students with a strong technical foundation in computer science, along with advanced domain knowledge in the field of cybersecurity.

Master of Science in Data Science

The MS in Data Science prepares students for advancement in professional careers. Data Scientists enable knowledge discovery in almost all of the subfields of science, social science, business, and policy. As businesses and government continue to turn to data-informed decision making, data scientists will become more necessary and influential within society as a whole. This program is designed to provide students with a breadth of knowledge in computer science, probability and statistics, data management and exploration, and machine learning, as well as the emerging technologies that will be gaining importance in the future.

Doctor of Philosophy in Computer Science

Degrees and GPA Requirements

Bachelors degree: All graduate applicants must hold an earned baccalaureate from a regionally accredited college or university or the recognized equivalent from an international institution.

University GPA requirement: The minimum grade point average for admission consideration for graduate study at the University of Denver must meet one of the following criteria:

- A cumulative 2.5 on a 4.0 scale for the baccalaureate degree.
- A cumulative 2.5 on a 4.0 scale for the last 60 semester credits or 90 quarter credits (approximately two years of work) for the baccalaureate degree.
- An earned master's degree or higher from a regionally accredited institution or the recognized equivalent from an international institution supersedes the minimum GPA requirement for the baccalaureate.
- A cumulative GPA of 3.0 on a 4.0 scale for all graduate coursework completed for applicants who have not earned a master's degree or higher.

Prerequisites:

- Prerequisite courses for the PhD include: COMP 1671 Introduction to Computer Science I, COMP 1672 Introduction to Computer Science II, COMP 2673 Introduction to Computer Science III, COMP 2300 Discrete Structures in Computer Science, COMP 2370 Introduction to Algorithms & Data Structures, and COMP 2691 Introduction to Computer Organization (or equivalent).

English Language Proficiency Test Score Requirements

The minimum TOEFL/IELTS/C1 Advanced/Duolingo English Test score requirements for this degree program are:

- Minimum TOEFL Score (Internet-based test): 80
- Minimum IELTS Score: 6.5
- Minimum C1 Advanced Score: 176
- Minimum Duolingo English Test Score: 115

Master of Science Applied Data Science and Artificial Intelligence

Degree and GPA Requirements

Bachelor's degree: All graduate applicants must hold an earned baccalaureate from a regionally accredited college or university or the recognized equivalent from an international institution.

University GPA requirement: The minimum grade point average for admission consideration for graduate study at the University of Denver must meet one of the following criteria:

- A cumulative 2.5 on a 4.0 scale for the baccalaureate degree.
- A cumulative 2.5 on a 4.0 scale for the last 60 semester credits or 90 quarter credits (approximately two years of work) for the baccalaureate degree.
- An earned master's degree or higher from a regionally accredited institution or the recognized equivalent from an international institution supersedes the minimum GPA requirement for the baccalaureate.
- A cumulative GPA of 3.0 on a 4.0 scale for all graduate coursework completed for applicants who have not earned a master's degree or higher.

English Language Proficiency Test Score Requirements

The minimum TOEFL/IELTS/C1 Advanced/Duolingo English Test score requirements for this degree program are:

- Minimum TOEFL Score (Internet-based test): 80
- Minimum IELTS Score: 6.5
- Minimum C1 Advanced Score: 176
- Minimum Duolingo English Test Score: 115

Master of Science in Computer Science

Degrees and GPA Requirements

Bachelors degree: All graduate applicants must hold an earned baccalaureate from a regionally accredited college or university or the recognized equivalent from an international institution.

University GPA requirement: The minimum grade point average for admission consideration for graduate study at the University of Denver must meet one of the following criteria:

- A cumulative 2.5 on a 4.0 scale for the baccalaureate degree.
- A cumulative 2.5 on a 4.0 scale for the last 60 semester credits or 90 quarter credits (approximately two years of work) for the baccalaureate degree.
- An earned master's degree or higher from a regionally accredited institution or the recognized equivalent from an international institution supersedes the minimum GPA requirement for the baccalaureate.
- A cumulative GPA of 3.0 on a 4.0 scale for all graduate coursework completed for applicants who have not earned a master's degree or higher.

Prerequisites

- Prerequisite courses for the MS include: COMP 1671 Introduction to Computer Science I, COMP 1672 Introduction to Computer Science II, COMP 2673 Introduction to Computer Science III, COMP 2300 Discrete Structures in Computer Science, COMP 2370 Introduction to Algorithms & Data Structures, and COMP 2691 Introduction to Computer Organization (or equivalent).

English Language Proficiency Test Score Requirements

The minimum TOEFL/IELTS/C1 Advanced/Duolingo English Test score requirements for this degree program are:

- Minimum TOEFL Score (Internet-based test): 80
- Minimum IELTS Score: 6.5
- Minimum C1 Advanced Score: 176
- Minimum Duolingo English Test Score: 115

Master of Science in Cybersecurity

Degrees and GPA Requirements

Bachelors degree: All graduate applicants must hold an earned baccalaureate from a regionally accredited college or university or the recognized equivalent from an international institution.

University GPA requirement: The minimum grade point average for admission consideration for graduate study at the University of Denver must meet one of the following criteria:

- A cumulative 2.5 on a 4.0 scale for the baccalaureate degree.
- A cumulative 2.5 on a 4.0 scale for the last 60 semester credits or 90 quarter credits (approximately two years of work) for the baccalaureate degree.
- An earned master's degree or higher from a regionally accredited institution or the recognized equivalent from an international institution supersedes the minimum GPA requirement for the baccalaureate.
- A cumulative GPA of 3.0 on a 4.0 scale for all graduate coursework completed for applicants who have not earned a master's degree or higher.

Prerequisites

- Applicants must have the prerequisite knowledge equivalent to an undergraduate minor or greater in computer science and are required to pass a computer science placement exam prior to matriculation into the graduate program. Students with deficiencies will be eligible to complete the foundational courses or pass the computer science placement exam prior to matriculation. The foundational courses are COMP 3003: Foundations in Computer Systems, COMP 3004: Foundational in Discrete Structures & Algorithms, and COMP 3005: Foundations in Python Programming.

English Language Proficiency Test Score Requirements

The minimum TOEFL/IELTS/C1 Advanced/Duolingo English Test score requirements for this degree program are:

- Minimum TOEFL Score (Internet-based test): 80
- Minimum IELTS Score: 6.5
- Minimum C1 Advanced Score: 176
- Minimum Duolingo English Test Score: 115

Doctoral Program

Doctor of Philosophy in Computer Science

Degree Requirements

Coursework Requirements

Code	Title	Credits
Three quarters minimum of COMP 4600 : Seminar in Computer Science		
At least 36 credits must be at the 4000-level courses		
Up to 24 credits may be taken in other relevant disciplines, as approved by the Computer Science Department Graduate Committee.		
Courses should be chosen in consultation with, and are subject to the approval of, the student's advisor.		
Total Credits		**90**

Minimum credit hours required: 90 beyond BA or BS degree

Additional degree requirements applicable to PhD students without a master's degree in Computer Science

- Must complete the requirements of the Master of Science in Computer Science with a thesis at a reasonable pace to remain on pace to complete the PhD in Computer Science on the expected timeline established by the advisor.

Additional Degree Requirements applicable to PhD Students with a 2-year master's degree in Computer Science or related field

- May take a proficiency test in the four required courses for master's degree (COMP 3351 Programming Languages, COMP 3361 Operating Systems I, COMP 3371 Data Structures & Algorithms and COMP 3200 Discrete Structures). The test may be offered at a time other than the official final exam time of the term. A grade of B+ (B plus) or better must be obtained in the test.
- If the student chooses not to take the proficiency test, the student must register and attend classes for the four required courses (COMP 3351 Programming Languages, COMP 3361 Operating Systems I, COMP 3371 Data Structures & Algorithms and COMP 3200 Discrete Structures). A grade of B+ (B plus) or better must be obtained in the courses.

Non-coursework Requirements:

- Written dissertation and oral defense that makes a significant contribution to the research literature in computer sciences
- Tool requirement
- Qualifying examination
- Preliminary examination

Qualifying & Dissertation Examinations
Qualifying Examination
Every PhD student must pass the qualifying exam. It consists of two parts, the breadth requirement and the written and oral exam.

1. Breadth Requirement: To fulfill the breadth requirement the student must take 5 graded courses (20 quarter credits) at the 3000- and 4000-level (not including independent study, internship, or independent research). At most, two may be at the 3000-level. At least three must be at the 4000-level. The course work should cover at least three distinct areas. Five areas should include a sequence of 3000- and 4000-level courses. The GPA in these courses must be at least 3.7/4.0. No course with a grade below a B may be used to fulfill this requirement. Graduate Computer Science courses taken at another university and transferred for credit at DU may be applied to the breadth requirement up to a maximum of 2 courses (8 quarter credits).
2. Written and Oral Exam: Before being admitted to this exam, the student must have fulfilled the breadth requirement.

The student selects an area of examination from the list of areas in Table 1. The written part of the exam is a take-home exam. It is a handed out on a Friday and is due the following Tuesday. The oral exam is held the following Friday. The take-home exam consists of a set of research questions, a set of related papers and instructions. The student should prepare a written report of at least 10 but no more than 20 pages with answers to the questions. Study guides or other relevant material to prepare for the exam can be obtained from the chair of the examination committee. The oral portion of the exam is based on a student presentation in which the student explains and defends his/her answers. During the oral exam, questions in other areas of computer science may also be asked.

A failed exam may be retaken once (in the same or another area). Sufficiently prior to the exam date, the department chair will appoint an examination committee of three tenure-track faculty. One of the committee members must be in the area in which the examination will be held. The student's advisor is allowed to be on the committee. The committee creates the take home exam and grades it. After the oral exam, the committee makes a recommendation to the Computer Science faculty on whether the student passes or fails. If the faculty agrees, the committee recommendation stands. If there is a disagreement, the faculty as a whole decides.

Preliminary Examination
Following successful completion of the Qualifying Examination, each student will prepare a dissertation proposal and take the preliminary examination. Passing this examination admits the student to Ph.D. candidacy. The dissertation proposal should be prepared in close consultation with the student's advisor and should be available to all committee members at least two weeks prior to the examination. It should reflect an extensive critical literature survey, and contain an accurate assessment of the state-of-the-art in the area of research, a precise statement of the problem to be solved, motivation for pursuing the research, and evidence to the effect that there is a good likelihood the problem is solvable with reasonable effort.

For full-time students, the preliminary examination must be taken within 5 quarters of passing the qualifying examination. Successful completion of the preliminary examination results in agreement between the student and the committee as to what will constitute successful completion of the dissertation research. The committee may choose to reconvene the examination to allow the student to further research the problem, complete additional course work, or revise the dissertation proposal document.

The examining committee consists of at least three Computer Science faculty members, including the advisor. The preliminary exam is a one hour oral closed exam. If a student passed the preliminary exam, but subsequently switches advisor and hence topic, the preliminary exam must be repeated within one year to ensure capability of the student and feasibility of the project.

Dissertation Defense
After the dissertation has been completed, the student must defend it in a final examination, as specified by the Office of Graduate Studies.

Tool Requirement
It is strongly recommended that students satisfy their tool requirement by demonstrating proficiency in a modern computer typesetting system suitable for writing technical papers that include mathematical equations and graphics. The faculty advisor must approve the specific system used to satisfy this requirement. Other options include reading competency in two languages selected from French, German, and Russian; a series of outside courses in another discipline; or significant laboratory experience involving computer science.

Master's Degree Programs

Master of Science in Applied Data Science and Artificial Intelligence

Degree Requirements

The MS in Applied Data Science and AI requires 48 credits. Students without prerequisite knowledge may take Algebra and Trigonometry (Math 1070) at DU prior to fully matriculating into the program. This program has 8 required classes (32 credits) and 4 electives (16 credits).

Code	Title	Credits
Required Courses		
COMP 3005	Foundations in Python Programming	4
COMP 3006	Python Software Development	4
COMP 3009	Essential Math for Data Science and AI	4
COMP 3421	Database Organization & Management I	4
COMP 4432	Machine Learning	4
COMP 4433	Data Visualization	4
COMP 4441	Introduction to Probability and Statistics for Data Science & AI	4
COMP 4531	Deep Learning: Model Design and Application	4
Elective Courses [1]		**16**
COMP 3904	Internship/Co-Op in Computing	
COMP 4334	Parallel and Distributed Computing for Data Science and AI	
COMP 4442	Advanced Probability and Statistics for Data Science and AI	
COMP 4449	Data Science & AI Capstone	
COMP 4450	Machine Learning Operations	
COMP 4451	Large Language Models for Data Science and AI	
COMP 4456	Deep Learning for Sequence Data	
COMP 4581	Algorithms for Data Science & Artificial Intelligence	
COMP 4711	Special Topics in Data Science and AI	
COMP 4991	Independent Study	
Other approved electives		
Total Credits		**48**

[1] Additional courses related to Data Science and AI may be approved by the department to count towards the 16 credit elective requirement. These might include topics courses such as Advanced Topics: Systems (COMP 3704) or COMP 3703 Topics-Artificial Intelligence.

Master of Science in Computer Science

Degree Requirements

Coursework Requirements

The Master of Science in Computer Science offers great flexibility in studying computer science beyond an undergraduate degree. Students without a background in computer science will typically begin with foundational courses that prepare them for more advanced work in computer science. An assessment is taken through the department that helps them determine where to begin.

Students are required to take 16 credits of required courses and 32 credits of electives. Of these 32 credits of elective computer science courses, 12 must be at the 4000-level, including 4 credits designated as "Theory". In addition, 8 of the 32 credits must be designated as advanced programming (and may also be 4000-level credits). Students wishing to pursue a thesis may apply up to 12 credits of Independent Research (COMP 4995) towards their 32 elective credits. Finally, students must take 3 quarters of Seminar in Computer Science (COMP 4600).

Code	Title	Credits
Foundational Courses [1]		**16**
COMP 3003	Foundations in Computer Systems	4
COMP 3004	Foundations in Discrete Structures & Algorithms	4
COMP 3005	Foundations in Python Programming	4
Required Courses		**16**
COMP 3200	Discrete Structures	4
COMP 3351	Programming Languages	4
COMP 3361	Operating Systems I	4
COMP 3371	Data Structures & Algorithms	4
Electives		**32**
Students must complete graduate-level electives to satisfy the following requirements.		
4000-level requirement		**12**
12 computer science elective credits at the 4000-level (excluding COMP 499X) are required, of which one must be designated as "Theory" (see below).		
Theory Requirement		**4**
4 credits of COMP at the 4000-level designated as "Theory". The current pre-approved list of theory courses includes, but is not limited to:		
COMP 4705	Advanced Topics-Programming (Advanced Programming Languages, Performance Modeling & Analysis)	
COMP 4591	Computational Geometry	
COMP 4372	Advanced Algorithms	
Advanced Programming Requirement		**8**
8 credits of COMP electives must include an advanced programming component. These courses must be approved by an advisor. The current pre-approved list includes but is not limited to:		
COMP 3352	Elements of Compiler Design	
COMP 3353	Compiler Construction	
COMP 3411	Web Programming II	
COMP 3432	Machine Learning	
COMP 3621	Computer Networking	
COMP 3801	Introduction Computer Graphics	
COMP 4510	Software for AI Robotics	
COMP 4621	Computer Networking	
Seminar Attendance Requirement		
COMP 4600	Seminar in Computer Science [2]	
Non-Thesis Option		
A maximum of 12 quarter hours may be earned in Independent Study (COMP 4991), provided the student can find an advisor for such independent study. No thesis is required. Not eligible for support (GTA, GRA).		
Thesis Option		

A maximum of 12 credits may be earned for thesis credits (COMP 4995 - Independent Research). A thesis of publishable quality and oral defense are required. A student receiving any support from the department (GTA, GRA) must complete the degree requirements as per the Thesis Option.	
Total Credits	**48-60**

1 Foundation Courses serve to provide a mathematical and programming background for students without a computer science undergraduate degree. Foundation course needs are determined by pre-assessments. Students may test out of one or more Foundation courses.

2 Thesis-track students must complete **three** quarters of COMP 4600 - Seminar (0 credits).
GTA's/GRA's are required to attend all seminars.

Outside Coursework

A maximum of 8 quarter hours may be earned in approved courses outside the COMP designation, including transfer credits from another university. Such credit must be approved in writing by an advisor from the computer science faculty.

Students should follow the rules and regulations stated in the departmental Graduate Student Manual.

Non-Coursework Requirements

Thesis Option

- Thesis and Oral Defense

Non-Thesis Option

- None

Master of Science in Cybersecurity

Degree Requirements

Coursework Requirements

Minimum credits required for degree: 48-60

Code	Title	Credits
Foundational Courses [1]		**0-12**
COMP 3003	Foundations in Computer Systems	4
COMP 3004	Foundations in Discrete Structures & Algorithms	4
COMP 3005	Foundations in Python Programming	4
Required Courses		**48**
COMP 3006	Python Software Development	4
COMP 3356	Introductory C/C++ Programming	4
COMP 3361	Operating Systems I	4
COMP 3731	Computer Forensics	4
COMP 4455	Shell Scripting and System Tools	4
COMP 4384	Secure Software Engineering	4
COMP 4621	Computer Networking	4
COMP 4721	Computer Security	4
COMP 4722	Network Security	4
COMP 4723	Ethical Hacking	4
COMP 4799	Capstone Project in Cybersecurity [2]	4
Cybersecurity Electives		**4**
Elective credits must be either a pre-approved cybersecurity course at DU, or with advisor approval. Pre-approved courses include COMP 4732, COMP 3421 and COMP 3424. Other courses with cyber security content at the discretion of the advisor or program director.		
Total Credits		48-60

1 Foundational Courses serve to provide a computer science foundation for students without a computer science background. Foundational Course needs are determined by pre-assessment by the department. Students may test out of one or more Foundational Courses.

2 Capstone Project in Cybersecurity (COMP 4799) is required; however, with advisor approval, 0 credit Internship/Co-Op in Computing (COMP 3904) may be used as the basis for the capstone course. If using the Internship/Co-Op option, students will register for 4799 normally, and apply through pioneercareers (PCO) for their internship.

No thesis is required.

Students should follow the rules and regulations stated in the departmental Graduate Student Manual.

GTA/GRA Support

Due to the intensive nature of this program, Cybersecurity students are not eligible for graduate teaching or research support.

Consult with Financial Aid at finaid@du.edu or at 303-871-4020 to discuss financial aid options.

Dual Undergraduate-Graduate Degree Programs

4+1 Dual degree BS in Computer Science and MS in Computer Science

The Department of Computer Science at the University of Denver offers a Dual Degree Bachelor of Science in Computer Science (http://bulletin.du.edu/undergraduate/majorsminorscoursedescriptions/traditionalbachelorsprogrammajorandminors/computerscience/#programofstudytext) and Master of Science in Computer Science (p.). The BS/MS combines the strong foundation of a BS in Computer Science and further depth and breadth in more advanced computer science topics through the MS in Computer Science.

Total Credit Hours: 183 at the undergraduate level (UG) for the Bachelor degree + 36 at the graduate level (GR) for the Master of Science degree = 219 credits

Undergraduate Requirements

Students who intend to obtain a BS/MS in Computer Science must satisfy all the requirements of the Bachelor of Science degree as outlined in the University of Denver Undergraduate Bulletin (http://bulletin.du.edu/undergraduate/), which includes completing two minors and all of the requirements for the Bachelor of Science in Computer Science (http://bulletin.du.edu/undergraduate/majorsminorscoursedescriptions/traditionalbachelorsprogrammajorandminors/computerscience/#programofstudytext) degree, including the Mathematics cognate requirements. In addition, 12 credits of the Breadth requirements for the BS in Computer Science are taken from the following list of **required** courses by the MS in Computer Science, but may be substituted for cross-listed COMP 3000-level undergraduate/graduate courses if these courses were taken prior to entry into the 4+1 program:

12 credits of the required MS in Computer Science courses that may be used for the Breadth requirement of the BS in Computer Science:		
COMP 3200	Discrete Structures	4
COMP 3351	Programming Languages	4
COMP 3361	Operating Systems I	4
COMP 3371	Data Structures & Algorithms	4

Graduate Requirements

The student must complete 36 credit hours at the graduate level of required coursework for the MS. These 36 hours include the **required** courses for the MS in Computer Science (the 4 classes listed above), unless they were taken as part of the Breadth requirement for the BS in Computer Science. In this case, other graduate-level courses will be substituted for them to meet the 36 credit hour requirement of the degree. Internship/Co-Op in Computing (COMP 3904) may not be used to substitute for these requirements.

Advanced Programming, Theory, and 4000-level Elective Requirements

As with the MS in Computer Science, students are also required to satisfy the Advanced Programming, Theory, and 4000-level elective requirements of the MS in Computer Science (p.) in their 36 credit hours of graduate coursework.

All electives, especially the MATH and COMP electives, should be selected in close consultation with an academic advisor from the Computer Science Department.

4:1 Dual Degree: BS in Computer Science and MS Data Science

The 4:1 dual degree enables DU students to earn a BS in Computer Science (CS) and an MS in Data Science (DS) in 5 years. A key benefit to the 4:1 student is the combination of disciplinary depth afforded by undergraduate work in CS, coupled with an applied, professional, focus provided by the DS program. Upon earning their BS degree, a 4:1 student will matriculate into the online track of the data science graduate program, enabling them to combine full-time employment with evening classes.

The MS in Data Science is a 48-credit program comprising 10 required courses and 2 electives and two admission tracks, online and on-campus. The online track provides maximum flexibility in course choice and timing. Students can matriculate into the program at the start of any term, with the full set of courses available to them. In contrast, the on-campus track is cohort-based; students start in the fall term and take a preset sequence of courses during their first year and part of the second. Due to these limitations, the 4:1 option will be limited to matriculation into the online MS in Data Science program track.

Sample schedule

The following is a sample schedule for the BS in Computer Science + MS in Computer Science. Note that other MS requirements would replace the Theory and Adv. Programming requirements, instead coming from the requirements of those degrees.

First Year					
Fall	**Credits**	**Winter**	**Credits**	**Spring**	**Credits**
First-year Seminar	4	WRIT 1122	4	WRIT 1133	4
COMP 1201	2	COMP 1202	2	COMP 1203	2
COMP 1351	3	COMP 1352	3	COMP 1353	3
Common Curriculum Courses	4	Math cognate	4	Math cognate	4
Math congnate	4	Common Curriculum Courses	4	Common Curriculum Course	4
	17		**17**		**17**
Second Year					
Fall	**Credits**	**Winter**	**Credits**	**Spring**	**Credits**
COMP 2300	4	COMP 2370	4	COMP 2361	4
COMP 2381	4	Common Curriculum Courses	4	Common Curriculum Courses	4
Common Curriculum Courses	4	Common Curriculum Courses	4	Common Curriculum Courses	4
Common Curriculum Courses	4			1st Minor Course	4
	16		**12**		**16**
Third Year					
Fall	**Credits**	**Winter**	**Credits**	**Spring**	**Credits**
COMP 2362	4	COMP 3xxx Elective	4	COMP 3xxx Elective	4
COMP 3xxx Elective	4	1st Minor Course	4	1st Minor Course	4
1st Minor Course	4	2nd Minor Course	4	2nd Minor Course	4
2nd Minor Course	4	Common Curriculum Courses	4	Common Curriculum Courses	4
	16		**16**		**16**
Fourth Year					
Fall	**Credits**	**Winter**	**Credits**	**Spring**	**Credits**
COMP 3351	4	COMP 3200	4	COMP 3371	4
COMP 3361	4	2nd Minor Course	4	Elective	4
1st Minor Course	4	Elective	4	Elective	4
2nd Minor Course	4	COMP 4xxx Grad Course	4	COMP 4xxx Grad Course	4
	16		**16**		**16**
Fifth Year					
Fall	**Credits**	**Winter**	**Credits**	**Spring**	**Credits**
COMP 4xxx Theory	4	COMP 4xxx Adv Programming	4	COMP 4xxx Adv Programming	4
COMP 4xxx Grad Course	4	COMP 4xxx Grad Course	4	COMP 4xxx Grad Course	4
COMP 4xxx Grad Course	4	COMP 4600	0	COMP 4600	0
COMP 4600	0				
	12		**8**		**8**

Total Credits: 219

COMP 3001 C and C++ Programming Foundations for New Graduate Students (4 Credits)
This accelerated course covers the basics of discrete mathematics including functions, relations, counting, logic, proofs etc that is necessary to attend CS graduate school. In addition, it includes an introduction to programming and algorithm analysis. Enrollment restricted to graduate students.

COMP 3002 C and C++ Foundations II for New Graduate Students (4 Credits)
This accelerated course continues to build on the basics of discrete mathematics by covering material including advanced counting, recurrences, graphs, trees, traversals, automata etc. that is necessary to attend Computer Science graduate school. In addition, it includes an introduction to additional algorithms and data structures. Prerequisite: COMP 3001.

COMP 3003 Foundations in Computer Systems (4 Credits)
This course introduces computer systems, including instruction set architectures; memory hierarchies including registers, caching, virtual memory, paging, and segmentation; number representations; binary arithmetic and operations; assembly language instructions; and pipelining in the CPU.

COMP 3004 Foundations in Discrete Structures & Algorithms (4 Credits)
Both discrete mathematics and an understanding of algorithms along with their analysis form principle foundations in computer science. In this course, the fundamentals of discrete mathematics including functions, relations, counting, logic, proofs, counting, recurrences, and probability are covered. In addition, beginning data structures and algorithms are covered including linked-lists, graphs, hash-tables, sorting, and binary search. An analysis of these data structures and algorithms is also covered through big-O notation and proof methods.

COMP 3005 Foundations in Python Programming (4 Credits)
This accelerated course covers the basics of Python programming. By the end of the course students will be able to develop, design and implement Python programs, explain the differences between data types, learn to read from and write to files, understand and use data structures, understand and use object oriented programming, and use Python packages.

COMP 3006 Python Software Development (4 Credits)
This accelerated course covers advanced Python programming for data scientists and cybersecurity professionals. Course Objectives: name and demonstrate proficiency using advanced Python programming techniques; analyze a programming task and create a development plan and high-level software design that accomplishes the task; relate common portions of the Python standard library to specific programming tasks; understand and apply aspects of the Python scientific programming ecosystem to achieve an analysis goal. Prerequisites: COMP 3005.

COMP 3007 Foundations in Data Science Mathematics I (4 Credits)
This course presents the elements of calculus essential for work in data science. Students will study differentiation and integration in the context of probability density and of optimization.

COMP 3008 Foundations in Data Science Mathematics II (4 Credits)
This course presents the elements of linear algebra and discrete math essential for subsequent coursework in data science.

COMP 3009 Essential Math for Data Science and AI (4 Credits)
The primary objective of the course is to empower students with the mathematical tools most used in data science. Students will develop an understanding of fundamental topics such as matrix operations, eigenvectors, and singular value decomposition (SVD), along with key calculus principles, including derivatives and integrals, with a focus on their applications in data science. The course also explores elements of multivariate calculus and optimization techniques to solve real-world problems, as well as probability concepts relevant to modeling uncertainty and variability in data. Additionally, students will gain exposure to implementing mathematical models using Python libraries such as NumPy, Pandas, and Matplotlib to analyze and manipulate data. By synthesizing these mathematical principles, students will be prepared for more advanced topics in data science and machine learning. Graduate student will be expected to upload a course presentation/project assignment to their individual Github portfolio.

COMP 3200 Discrete Structures (4 Credits)
Discrete mathematical structures and non-numerical algorithms; graph theory, elements of probability, propositional calculus, Boolean algebras; emphasis on applications to computer science. Prerequisites: COMP 2300 and COMP 1353 for undergraduates and COMP 3004 for graduates.

COMP 3351 Programming Languages (4 Credits)
Learn the fundamentals of programming languages through functional programming through an in-depth understanding of syntax and semantics around program structures and how programming languages are parsed and interpreted. Understand recursion as a fundamental problem-solving paradigm and the important role that higher order types and kinds play in eliminating errors and simplifying software development. Satisfies the Programming Language elective requirement. Prerequisites: COMP 2362 and COMP 2370.

COMP 3352 Elements of Compiler Design (4 Credits)
Techniques required to design and implement a compiler; topics include lexical analysis, grammars and parsers, type-checking, storage allocation and code generation. Prerequisite: COMP 3351.

COMP 3353 Compiler Construction (4 Credits)
Design and implementation of a major piece of software relevant to compilers. Prerequisite: COMP 3352.

COMP 3356 Introductory C/C++ Programming (4 Credits)
This course introduces unmanaged programming language concepts to students whose primary programming experience is in a managed language (Java/Python, etc.). Concepts like type safety, manual memory management and “unsafe” library functions are covered. Common pitfalls in these languages from which most security issues arise are explained and students gain experience in understanding such code and evaluating it for program errors. Students will also be introduced to important compiled language concepts of static/dynamic linking, compilation and debugging. Prerequisites: COMP 3006.

COMP 3361 Operating Systems I (4 Credits)
Operating systems functions and concepts; processes, process communication, synchronization; processor allocation, memory management in multiprogramming, time sharing systems. Prerequisites: for undergraduates: (COMP 2355 and COMP 2691) or COMP 2361; COMP 2370; for graduate students: COMP 3003, 3004, and 3005.

COMP 3371 Data Structures & Algorithms (4 Credits)
Design and analysis of algorithms and data structures; asymptotic complexity, recurrence relations, lower bounds; algorithm design techniques such as incremental, divide-and-conquer, dynamic programming, iterative improvement, greedy algorithms; randomized data structures and algorithms. Prerequisites: COMP 2370 or equivalent and COMP 3200.

COMP 3372 Advanced Algorithms (4 Credits)
Advanced techniques for the design and analysis of algorithms and data structures; amortized complexity, self-adjusting data structures; randomized , online, and string algorithms; NP-completeness, approximation and exact exponential algorithms; flow networks.

COMP 3381 Software Engineering I (4 Credits)
An introduction to software engineering. Topics include software processes, requirements, design, development, validation and verification and project management. Cross-listed with COMP 4381. Prerequisites: COMP 3351; COMP 3361 or COMP 2362; or instructor permission.

COMP 3382 Software Engineering II (4 Credits)
Continuation of COMP 3381. Topics include component-based software engineering, model-driven architecture, and service-oriented architecture. Prerequisite: COMP 3381.

COMP 3411 Web Programming II (4 Credits)
In this course you will learn how to develop a full-stack web application that is capable of serving dynamic content from a database. Furthermore, you will learn the core design concepts and principles that will enable you to develop scalable and easy to maintain webapplications - a set of skills that will serve you well in both your personal and professional projects in the future. Prerequisite: COMP 3410.

COMP 3412 Web Projects: Web Development III (4 Credits)
In this course you will learn how to develop, as a group, a full-stack web application that is capable of serving dynamic content from a database. We will use the MongoDB, ExpressJS, Angular, and Node.js (MEAN) software stack to work on a real-life problem presented to us by an external product owner. In the class we will use the Scrum framework for Agile development to work, as a software team, through several sprints of development. You will be peer reviewing each other throughout the course, and the product owner will also be reviewing your product through end-of-sprint demos as features are completed. The goal for this class is for it to be a fun, collaborative, and educational environment that demonstrates what it is like to work as a real software team. Prerequisite: COMP 3411.

COMP 3421 Database Organization & Management I (4 Credits)
An introductory class in database management systems covering both relational and non-relational databases with an emphasis on relational. Topics include database design, ER modeling, relational algebra, SQL, scripting, and embedded SQL. Each student will design, load, query and update a nontrivial database using a relational database management system (RDBMS). In addition, an introduction to a NoSQL database will be included. Graduate students will read one or two relevant technical papers and write a summary report. Prerequisites: for undergraduates: COMP 1353 or COMP 2673; for graduates: COMP 3005 or COMP3006 (MS Data Science).

COMP 3424 NoSQL Databases (4 Credits)
In this course, students learn what NoSQL databases are, learn to identify the differences between them, and gain a fundamental understanding between SQL, relational databases, and NoSQL databases. Students further explore which type of NoSQL database is the correct one given a use-cases, examining types, methods of communicating with it, contrasts to other NoSQL databases, performance and scalability. Prerequisites: for undergraduates, COMP 2355 or COMP 2361; for graduates: COMP 3005.

COMP 3431 Data Mining (4 Credits)
Data Mining is the process of extracting useful information implicitly hidden in large databases. Various techniques from statistics and artificial intelligence are used here to discover hidden patterns in massive collections of data. This course is an introduction to these techniques and their underlying mathematical principles. Topics covered include: basic data analysis, frequent pattern mining, clustering, classification, and model assessment. Prerequisites: COMP 2370.

COMP 3621 Computer Networking (4 Credits)
An introduction to computer networks with an emphasis on Internet protocols. Topics include: internet design, application layer protocols such as SMTP and HTTP, session layer protocols including TCP and UDP, the internet protocol (IP), link layer technology such as Ethernet, and security issues related to networking. Programming experience of client/server architectures using sockets and TCP/UDP through projects is emphasized. Prerequisites: for undergraduates: (COMP 2355 or COMP 2361) and COMP 2370; for graduates COMP 3004 and COMP 3006. Cross listed with COMP 4621.

COMP 3681 Networking for Games (4 Credits)
Implementing the networking code for multiplayer games is a complex task that requires an understanding of performance, security, game design, and advanced programming concepts. In this course, students are introduced to the networking stack and how this is connected to the Internet, learn how to write protocols for games, and implement several large games using a game engine that demonstrate the kind of networking and protocols required by different genres of games. In addition, tools are introduced that help understand and debug networking code, simplify the creation of protocols, and make the development of networking code easier.

COMP 3701 Topics in Computer Graphics (4 Credits)

COMP 3702 Topics in Database (4 Credits)

COMP 3703 Topics-Artificial Intelligence (4 Credits)

COMP 3704 Advanced Topics: Systems (4 Credits)

COMP 3705 Topics in Computer Science (1-4 Credits)

COMP 3731 Computer Forensics (4 Credits)
Computer Forensics involves the examination of information contained in digital media with the aim of recovering and analyzing latent evidence. This course will provide students an understanding of the basic concepts in preservation, identification, extraction and validation of forensic evidence in a computer system. The course covers many systems level concepts such as disk partitions, file systems, system artifacts in multiple operating systems, file formats, email transfers, and network layers, among others. Students work extensively on raw images of memory and disks, and in the process, build components commonly seen as features of commercial forensics tools (e.g. file system carver, memory analyzer, file carver, and steganalysis). Prerequisites: COMP 3361; COMP 2355 or 2361 for undergraduates; COMP 3006 for graduates.

COMP 3801 Introduction Computer Graphics (4 Credits)
Fundamentals of 3D rendering including the mathematics behind coordinate systems, projections, clipping, hidden surface removal, shadows, lighting models, shading models, and mapping techniques. Significant use of 3D APIs through shader programming is covered along with the basics of 3D model representation and animations. Satisfies "Advanced Programming" requirements for graduate students. Prerequisites: COMP 2370, MATH 1952 or 1962.

COMP 3820 Introduction to Game Programming (4 Credits)
Learn the fundamentals of game programming by creating 2D and 3D games using a modern game engine. Topics include working with 2D/3D art assets, character controllers, physics, camera systems, 2D and 3D animation integration, lighting, audio, user interfaces, artificial intelligence, and level design. The emphasis in this course is on game programming and implementing game mechanics in fully-working games. Prerequisites: COMP 2381 and COMP 2821.

COMP 3821 Game Programming I (4 Credits)
A continuation of Introduction to Game Programming, this course introduces advanced topics that are essential as future game programmers. Students have the opportunity to learn game engine architecture, 2D and 3D linear algebra for graphics, sprites, 2D and 3D animations, input handling, finite state machines, particle systems, user interfaces, game audio, and artificial intelligence for games. Prerequisites: COMP 2370 and COMP 3820.

COMP 3822 Game Programming II (4 Credits)
In this course, students learn how to work with a 3D game engine and build 3D games. Topics include algorithms, mathematics for 3D game engines, scene management, animations, 3D shaders, particle systems, physics for games, UIs, terrain systems, and working with higher-level scripting languages on top of the low-level implementation language. Prerequisites: COMP 3821. Suggested corequisite or prerequisite: COMP 3801.

COMP 3904 Internship/Co-Op in Computing (0-10 Credits)
Experiential learning through employment with a company to work in computer science. Students are expected to find the internship/co-op and connect their hiring manager with an advisor in the department to get approval. Requirements for approval include a job description that shows what will be accomplished throughout the quarter and a final report from the hiring manager to report on their performance so that a grade may be assigned. Prerequisites: COMP 2370 and approval of department. MS Data Science students: practical experience working under the supervision of data science employer and a data science faculty advisor. Learning outcomes will be assessed via student work such as summary reports, literature reviews, presentations, code notebooks and/or repositories. Prerequisites: permission of Data Science Faculty Director.

COMP 3991 Independent Study (1-10 Credits)
Cannot be arranged for any course that appears in the regular course schedule for that particular year.

COMP 4100 Human-Computer Interaction (4 Credits)
Introduces students in computer science and other disciplines to principles of and research methods in human-computer interaction (HCI). HCI is an interdisciplinary area concerned with the study of interaction between humans and interactive computing systems. Research in HCI looks at cognitive and social phenomena surrounding human use of computers with the goal of understanding their impact and creating guidelines for the design and evaluation of software, interfaces, physical products, and services in industry. No prerequisites are required to take the course and students from all disciplines are welcome. Cross listed with COMP 3100.

COMP 4333 Parallel and Distributed Computing (4 Credits)
Current techniques for effective use of parallel processing and large scale distributed systems. Programming assignments will give students experience in the use of these techniques. Specific topics will vary from year to year to incorporate recent developments. This course qualifies for the Computer Science "Advanced Programming" requirement. Prerequisites: COMP2370 and COMP2355, or equivalent.

COMP 4334 Parallel and Distributed Computing for Data Science and AI (4 Credits)
Current techniques for effective use of parallel processing and large-scale distributed systems for data science. Programming assignments will give students experience in the use of these techniques. Specific topics will vary from year to year to incorporate recent developments. This course is not to be used for the MS Computer Science. Graduate students are required to contribute at least one course assignment to their personal Github portfolio. Prior completion or concurrent enrollment in COMP 4432 is strongly recommended. Prerequisite: COMP 3006.

COMP 4355 Advanced System Programming (4 Credits)
This course covers programming in a UNIX environment, including use of common command line utilities, scripting, source control via Git, and integration of POSIX system calls into C/C++ code. These features will be leveraged to solve practical problems cleanly and efficiently. More emphasis will be placed on using these features than on how those features work. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4370 Algorthmic Problem Solving (4 Credits)
The course is intended for students who are familiar with programming syntax but have not had much experience writing computer programs to solve a problem stated as a high-level description. The course will run through multiple such problem descriptions, discuss the design of programs to solve those problems using popular data structures, and have students implement those designs using a programming language. This course does not count for MS Computer Science requirements. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4372 Advanced Algorithms (4 Credits)
Advanced techniques for the design and analysis of algorithms and data structures; amortized complexity, self-adjusting data structures; randomized , online, and string algorithms; NP-completeness, approximation and exact exponential algorithms; flow networks. Prerequisite: COMP 3371. Cross listed with COMP 3372.

COMP 4384 Secure Software Engineering (4 Credits)
This course is concerned with systematic approaches for the design and implementation of secure software. While topics such as cryptography, networking, network protocols and large scale software development are touched upon, this is not a course on those topics. Instead, this course is on identification of potential threats and vulnerabilities early in the design cycle. The emphasis in this course is on methodologies and paradigms for identifying and avoiding security vulnerabilities, formally establishing the absence of vulnerabilities, and ways to avoid security holes in new software. There are programming assignments designed to make students practice and experience secure software design and development. Prerequisites: COMP 3006, COMP 3361, COMP 3356.

COMP 4401 Introduction to Python for Data Science (2 Credits)
This eight-week course enables aspiring graduate students in data science to meet their Python programming proficiency prior to matriculation. Students who complete this course will be able to write and modularize code, design efficient control structures, perform basic data operations, debug errors, distinguish data types and basic structures, and work within an integrated development environment.

COMP 4431 Data Mining (4 Credits)
Data Mining is the process of extracting useful information implicitly hidden in large databases. Various techniques from statistics and artificial intelligence are used here to discover hidden patterns in massive collections of data. This course is an introduction to these techniques and their underlying mathematical principles. Topics covered include: basic data analysis, frequent pattern mining, clustering, classification, and model assessment. Prerequisites: COMP 4441 and COMP 4581.

COMP 4432 Machine Learning (4 Credits)
This course will give an overview of traditional machine learning techniques, their strengths and weaknesses, and the problems they are designed to solve. This will include the broad differences between supervised/unsupervised and reinforcement learning as well as associated learning problems such as classification and regression. Techniques covered, at the discretion of the instructor, may include approaches such as linear and logistic regression, support vector machines, kNN, decision trees, random forests, Naive Bayes, EM, k-Means, and PCA. After course completion, students will have a working knowledge of these approaches and experience applying them to learning problems. Graduate students will contribute and assignment or project to a digital portfolio on Github. Prerequisites: MATH 3009 and COMP 3006.

COMP 4433 Data Visualization (4 Credits)
This course explores visualization techniques and theory. The course covers how to use visualization tools to effectively present data as part of quantitative statements within a publication/report and as an interactive system. Both design principles (color, layout, scale, and psychology of vision) as well as technical visualization tools/languages will be covered. Graduate students will contribute one assignment or project to a personal portfolio on Github. Prerequisite: COMP 3006.

COMP 4441 Introduction to Probability and Statistics for Data Science & AI (4 Credits)
The course introduces fundamentals of probability for data science. Students survey data visualization methods and summary statistics, develop models for data, and apply statistical techniques to assess the validity of the models. The techniques will include parametric and nonparametric methods for parameter estimation and hypothesis testing for a single sample mean and two sample means, for proportions, and for simple linear regression. Students will acquire sound theoretical footing for the methods where practical, and will apply them to real-world data, primarily using R. Graduate students will contribute a course assignment or project to their personal portfolio in Github. Prerequisite: COMP 3009.

COMP 4442 Advanced Probability and Statistics for Data Science and AI (4 Credits)
This course builds on material in Probability and Statistics 1. Students will carry out model fitting and diagnostics for multiple regression, ANOVA, ANCOVA, and generalized linear models. Dimension reductions techniques such as PCA and Lasso are introduced, as are techniques for handling dependent data. The course introduces the principles of resampling and Bayesian Analysis. Students will acquire sound theoretical footing for the methods where practical, and will apply them to real-world data, primarily using R. Enforced Prerequisites: COMP 4441.

COMP 4447 Data Science Tools 1 (4 Credits)
Organizations are using data science to extract actionable insight from data. To highlight the hidden patterns in the data, this course equips students with essential sills for data collection, cleanup, transformation, feature engineering, summarization, visualization, and collaboration. Students will do assignments and a final project. This is a hands-on course. Students will use Python libraries, Linux commands, API requests, web-scraping, GitHub and various data sets to perform these activities. Prerequisites: COMP 3006.

COMP 4448 Data Science Tools 2 (4 Credits)
Building a successful predictive model is a multi-faceted process. This course focuses on hypothesis testing and the development of predictive models. Students will learn the data science process for model development while learning to program a variety of predictive algorithms. Students will do assignments and a final project. This is a hands-on course. Students will use Python libraries, Linux commands, and various data sets to perform these activities. Prerequisite: COMP 4447.

COMP 4449 Data Science & AI Capstone (4 Credits)
Data Science Capstone provides students an opportunity to demonstrate their expertise as data scientists. Students are expected to integrate prior knowledge and skills to design, develop, test, and present 'full-cycle' data science products, and apply them in real-world contexts. This includes assessing and communicating their value to decision-making. Two sets of challenges of increasing complexity are presented as mid-term and term projects. These two projects are implemented, documented, tested, and presented by the students or students' two-person teams. Prerequisites: COMP 4432.

COMP 4450 Machine Learning Operations (4 Credits)
This course introduces the engineering and deployment of machine learning systems. Students will learn essential technical and organizational skills needed to develop and deploy ML solutions, progressing from fundamental concepts through to practical implementation. Key topics include ML project lifecycle management, containerization, web application development, data management strategies, and cloud deployment. Focus will be given to the application of industry-standard tools and best practices for building production-ready machine learning systems. Prerequisites: COMP 3421 and COMP 4432.

COMP 4455 Shell Scripting and System Tools (4 Credits)
This course covers navigating and utilizing tools in a UNIX environment, including use of common command line utilities, Bash and Python shell scripting, source control via Git, pipes and I/O redirection, networking in Python and OS multi-processing/multi-threading. More emphasis will be placed on using these tools than on how those tools work. Prerequisite: COMP 3006.

COMP 4456 Deep Learning for Sequence Data (4 Credits)
This course introduces modern approaches to analyzing sequential data, with a focus on deep learning methods for time series and natural language processing. Students will develop practical skills in building and implementing neural network models for sequential data analysis, while gaining a thorough understanding of the underlying concepts. Through hands-on projects and real-world applications, students will learn to process, analyze, and generate predictions from various types of sequential data. The course emphasizes practical implementation and modern deep learning frameworks, bridging the gap between traditional statistical methods and contemporary neural network approaches. Prerequisite: COMP 4432.

COMP 4490 Understanding AI (4 Credits)
Understanding Artificial Intelligence provides a comprehensive introduction to artificial intelligence (AI) for students of all backgrounds. This course explores the diverse approaches to AI, including rule-based systems, expert systems, knowledge representation, search algorithms, machine learning, natural language processing, robotics, and human-AI interaction. Students will gain hands-on experience with AI tools and applications while critically examining their impact on society. The course emphasizes the capabilities, limitations, and ethical considerations of AI, preparing students to engage thoughtfully with AI technologies in various fields.

COMP 4491 Applications of Generative AI (4 Credits)
Applications of Generative Artificial Intelligence is a hands-on course explores the transformative power of generative artificial intelligence (AI) across various industries. Students will engage directly with generative AI tools and techniques, including text generation, image synthesis, music composition, code generation, and more. Through practical exercises, projects, and case studies, students will develop a deep understanding of how generative AI can enhance creativity, automate workflows, and drive innovation in diverse fields. This course is designed for students who have completed COMP 4490 and want to apply AI in practical, industry-relevant contexts. Prerequisite: COMP 4490 and non-majors.

COMP 4495 AI Design Project (4 Credits)
The Artificial Intelligence Design Project serves as the culminating experience for students who have completed COMP 4490 and COMP 4491. This course provides an opportunity for students to apply their AI knowledge and skills to a self-directed project in a domain of their choice. Over ten weeks, students will identify a problem, design an AI-driven solution, and develop a functional prototype or proof-of-concept. Emphasizing creativity, problem-solving, and ethical considerations, this capstone-style course enables students to make a tangible impact in their field. The course includes milestone check-ins, peer reviews, and a final presentation to showcase their work. Pre-requisite: COMP 3490 and COMP 3491.

COMP 4501 Introduction to Artificial Intelligence (4 Credits)
Introduces a variety of Artificial Intelligence concepts and techniques, relevant to a broad range of applications. Students survey multiple techniques including search, knowledge representation and reasoning, probabilistic inference, machine learning, and natural language processing. Examines concepts of constraint programming, evolutionary computation and non-standard computation. Graduate students will also consider current political, economic, and societal issues related to artificial intelligence through reading and discussion. Prerequisites: COMP 3004 or equivalent.

COMP 4525 Adv. Human Centered AI - Mind Reading Machines (4 Credits)
This course explores a key intersection of Artificial Intelligence (AI) and Human-Computer Interaction (HCI). How can AI enhance our interactions with computers by catering to our individual needs and differences? Can we teach computers to understand people's thoughts and feelings to improve overall interaction? Humans often adjust their communication when they notice the person they're talking to is frustrated—why can't computers do the same? In this interdisciplinary, research-focused course, you will read, present, and discuss seminal papers at the intersection of AI and HCI, including (but not limited to) affective computing, physiological computing, augmented cognition, and multimodal interaction. ## Learning Objectives (General Goal) Demonstrate a broad understanding of the intersections between Human-Computer Interaction and Artificial Intelligence, with an emphasis on user modeling. (LO1) Recognize how theories from fields outside of Computer Science can inform and enhance Computer Science practices. (LO2) Develop knowledge of state-of-the-art user modeling techniques, including practical applications and associated challenges. (LO3) Develop critical thinking and scientific communication skills related to current AI topics, including science writing and scientific discourse. (LO4 - Graduate Students Only) Consider how the intersection of Humans and Artificial Intelligence impacts your own research and work. Consider the ethical challenges of your own work more deeply and how these may be mitigated.

COMP 4531 Deep Learning: Model Design and Application (4 Credits)
This course addresses the foundational concepts and components of Artificial Neural Networks (ANN), highlighting their capabilities, strengths, and weaknesses as a machine learning algorithm. Students taking this course will develop ANN models from scratch in Python as a basis for understanding their design as well as the underlying mechanics and calculations that shape their behavior. Key topics such as forward-backward propagation, loss function characteristics and optimization will be considered in relation to model design and computational efficiency as well as to problems such as exploding and vanishing gradients. Training strategies (e.g., dropout, initialization, batch normalization) will further enable students to assess trade-offs in model bias & variance. Coupled with hands-on assignments, these building blocks provide the knowledge and skills required to effectively design and implement ANN models that are ethically and technically sound. As well as foreground important architectures such as Convolutional ANNs, Recurrent ANNs, LSTMS, and Transformers as well as their applicability to modern problems. Student learning and proficiency will be assessed based on a combination of quizzes, coding assignments, exams, and a culminating project. Prerequisite: COMP 4432.

COMP 4581 Algorithms for Data Science & Artificial Intelligence (4 Credits)
This course introduces the design and analysis of algorithms within the context of data science and artificial intelligence. Topics include; asymptotic complexity and algorithm design techniques such as incremental, divide and conquer, dynamic programming, randomization, greedy algorithms, and advanced sorting techniques. Examples to illustrate techniques are drawn from multi-dimensional clustering (k-means and probabilistic), regression, decision trees, order statistics, data mining using apriori algorithms, and algorithms for generating combinatorial objects. Graduate students will contribute at least 1 assignment to their personal portfolio on Github. Prerequisite: COMP 3006.

COMP 4591 Computational Geometry (4 Credits)
This class deals with the design and implementation of efficient algorithms for problems defined over geometric objects, such as points, lines, polygons, surfaces, etc. The methods and algorithms covered find applications in many areas, including computer graphics (e.g., hidden surface removal), computer-aided design and manufacturing (e.g., 3D printing), machine learning (e.g., supervised and unsupervised classification), geographic information systems (e.g. terrain visibility), robotics (e.g., motion planning), data mining (e.g., dimensionality reduction), and computer vision (3D reconstruction), to name a few. Fundamental geometric problems such as partitioning, proximity, intersection, convexity, visibility, point location, and motion planning are focused on. Efficient data structures and algorithms for their solutions and design techniques germane to the field, such as divide-and-conquer, plane sweep, randomization, duality, etc. are discussed in detail. Practical methods for the robust implementation of geometric algorithms are also covered. Prerequisites: COMP 3200 and COMP 3371. This course satisfies the Theory requirement for graduate students.

COMP 4600 Seminar in Computer Science (0-4 Credits)
Preparation and presentation of lectures on some aspect of current research in computer science; topics not generally encountered in formal courses, may include robotics, pattern recognition, parallel processing, computer applications. 10- to 15- page paper with bibliography required.

COMP 4621 Computer Networking (4 Credits)
The Internet is arguably the most transformative invention in recent history and is at its core a massive global computer network (of networks). Students in this course learn how the Internet works, from the highest-level application layer to the lowest-level hardware layer. Topics covered include the OSI and TCP/IP reference models, physical transmission methods, error detection and correction, addressing, routing algorithms, congestion control and more. Prerequisites: COMP 3006, COMP 3361 (or instructor approval). Cross listed with COMP 3621.

COMP 4701 Special Tpcs-Computer Graphics (1-4 Credits)

COMP 4702 Advanced Topics-Database (3 Credits)

COMP 4703 Adv Topics-Artificial Intell (1-4 Credits)

COMP 4704 Advanced Topics-Systems (3-4 Credits)

COMP 4705 Advanced Topics-Programming (1-4 Credits)

COMP 4709 Special Tpcs-Computer Security (1-4 Credits)

COMP 4711 Special Topics in Data Science and AI (4 Credits)
Special Topics in Data Science highlights selected methods & applications in machine learning that are not specifically addressed within the curriculum. Areas of study may include, but are not limited to, natural language processing, recommender systems, data science for social justice, machine learning operations, generative AI and more. Special Topics in Data Science will be offered as an elective on an intermittent basis. Prerequisites: COMP 4432 and COMP 4441.

COMP 4721 Computer Security (4 Credits)
This course gives students an overview of computer and system security along with some cryptography. Some network security concepts are also included. Other concepts include coverage of risks and vulnerabilities, policy formation, controls and protection methods, role-based access controls, database security, authentication technologies, host-based and network-based security issues. Prerequisites: COMP 3006, COMP 3361 (or advisor/instructor approval).

COMP 4722 Network Security (4 Credits)
Network Security covers tools and techniques employed to protect data during transmission. It spans a broad range of topics including authentication systems, cryptography, key distribution, firewalls, secure protocols and standards, and overlaps with system security concepts as well. This course will provide an introduction to these topics, and supplement them with hands-on experience. In addition, students will perform an extensive analysis, or development of a security related product independently. Prerequisites: COMP 4721.

COMP 4723 Ethical Hacking (4 Credits)
Ethical hacking is the process of probing computer systems for vulnerabilities and exposing their presence through proof-of-concept attacks. The results of such probes are then utilized in making the system more secure. This course will cover the basics of vulnerability research, foot printing targets, discovering systems and configurations on a network, sniffing protocols, firewall hacking, password attacks, privilege escalation, rootkits, social engineering attacks, web attacks, and wireless attacks, among others. Prerequisites: COMP 3361, or COMP 3001, 3002, 3003, and 3004.

COMP 4724 Systems Security Management (4 Credits)
This course covers basic system administration tasks on a Unix environment, with a special focus on command line navigation, file/process access control, setting up network configurations, and managing services related to networks and their security. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4799 Capstone Project in Cybersecurity (1-8 Credits)
The purpose of the cybersecurity capstone project is to provide an integrative experience that ties together the learning outcomes from academic coursework undertakings and industry skills necessary to be productive in delivering an end product. Students will engage in one of many options available, such as involvement in a research project, a case study, a product development project, or an extensive survey paper. Capstone projects are presented at the end of the quarter in front of a representative group. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4991 Independent Study (1-12 Credits)
Independent study on a particular topic supervised by a faculty member. Student must find a faculty member who will supervise work on the topic. Cannot be arranged for any course that appears in regular course schedule for that particular year.

COMP 4995 Independent Research (1-17 Credits)
Research projects undertaken in conjunction with a faculty member.

COMP 5991 Independent Study (1-17 Credits)

COMP 5995 Independent Research (1-17 Credits)

Courses

COMP 3001 C and C++ Programming Foundations for New Graduate Students (4 Credits)
This accelerated course covers the basics of discrete mathematics including functions, relations, counting, logic, proofs etc that is necessary to attend CS graduate school. In addition, it includes an introduction to programming and algorithm analysis. Enrollment restricted to graduate students.

COMP 3002 C and C++ Foundations II for New Graduate Students (4 Credits)
This accelerated course continues to build on the basics of discrete mathematics by covering material including advanced counting, recurrences, graphs, trees, traversals, automata etc. that is necessary to attend Computer Science graduate school. In addition, it includes an introduction to additional algorithms and data structures. Prerequisite: COMP 3001.

COMP 3003 Foundations in Computer Systems (4 Credits)
This course introduces computer systems, including instruction set architectures; memory hierarchies including registers, caching, virtual memory, paging, and segmentation; number representations; binary arithmetic and operations; assembly language instructions; and pipelining in the CPU.

COMP 3004 Foundations in Discrete Structures & Algorithms (4 Credits)
Both discrete mathematics and an understanding of algorithms along with their analysis form principle foundations in computer science. In this course, the fundamentals of discrete mathematics including functions, relations, counting, logic, proofs, counting, recurrences, and probability are covered. In addition, beginning data structures and algorithms are covered including linked-lists, graphs, hash-tables, sorting, and binary search. An analysis of these data structures and algorithms is also covered through big-O notation and proof methods.

COMP 3005 Foundations in Python Programming (4 Credits)
This accelerated course covers the basics of Python programming. By the end of the course students will be able to develop, design and implement Python programs, explain the differences between data types, learn to read from and write to files, understand and use data structures, understand and use object oriented programming, and use Python packages.

COMP 3006 Python Software Development (4 Credits)
This accelerated course covers advanced Python programming for data scientists and cybersecurity professionals. Course Objectives: name and demonstrate proficiency using advanced Python programming techniques; analyze a programming task and create a development plan and high-level software design that accomplishes the task; relate common portions of the Python standard library to specific programming tasks; understand and apply aspects of the Python scientific programming ecosystem to achieve an analysis goal. Prerequisites: COMP 3005.

COMP 3007 Foundations in Data Science Mathematics I (4 Credits)
This course presents the elements of calculus essential for work in data science. Students will study differentiation and integration in the context of probability density and of optimization.

COMP 3008 Foundations in Data Science Mathematics II (4 Credits)
This course presents the elements of linear algebra and discrete math essential for subsequent coursework in data science.

COMP 3009 Essential Math for Data Science and AI (4 Credits)
The primary objective of the course is to empower students with the mathematical tools most used in data science. Students will develop an understanding of fundamental topics such as matrix operations, eigenvectors, and singular value decomposition (SVD), along with key calculus principles, including derivatives and integrals, with a focus on their applications in data science. The course also explores elements of multivariate calculus and optimization techniques to solve real-world problems, as well as probability concepts relevant to modeling uncertainty and variability in data. Additionally, students will gain exposure to implementing mathematical models using Python libraries such as NumPy, Pandas, and Matplotlib to analyze and manipulate data. By synthesizing these mathematical principles, students will be prepared for more advanced topics in data science and machine learning. Graduate student will be expected to upload a course presentation/project assignment to their individual Github portfolio.

COMP 3200 Discrete Structures (4 Credits)
Discrete mathematical structures and non-numerical algorithms; graph theory, elements of probability, propositional calculus, Boolean algebras; emphasis on applications to computer science. Prerequisites: COMP 2300 and COMP 1353 for undergraduates and COMP 3004 for graduates.

COMP 3351 Programming Languages (4 Credits)
Learn the fundamentals of programming languages through functional programming through an in-depth understanding of syntax and semantics around program structures and how programming languages are parsed and interpreted. Understand recursion as a fundamental problem-solving paradigm and the important role that higher order types and kinds play in eliminating errors and simplifying software development. Satisfies the Programming Language elective requirement. Prerequisites: COMP 2362 and COMP 2370.

COMP 3352 Elements of Compiler Design (4 Credits)
Techniques required to design and implement a compiler; topics include lexical analysis, grammars and parsers, type-checking, storage allocation and code generation. Prerequisite: COMP 3351.

COMP 3353 Compiler Construction (4 Credits)
Design and implementation of a major piece of software relevant to compilers. Prerequisite: COMP 3352.

COMP 3356 Introductory C/C++ Programming (4 Credits)
This course introduces unmanaged programming language concepts to students whose primary programming experience is in a managed language (Java/Python, etc.). Concepts like type safety, manual memory management and “unsafe” library functions are covered. Common pitfalls in these languages from which most security issues arise are explained and students gain experience in understanding such code and evaluating it for program errors. Students will also be introduced to important compiled language concepts of static/dynamic linking, compilation and debugging. Prerequisites: COMP 3006.

COMP 3361 Operating Systems I (4 Credits)
Operating systems functions and concepts; processes, process communication, synchronization; processor allocation, memory management in multiprogramming, time sharing systems. Prerequisites: for undergraduates: (COMP 2355 and COMP 2691) or COMP 2361; COMP 2370; for graduate students: COMP 3003, 3004, and 3005.

COMP 3371 Data Structures & Algorithms (4 Credits)
Design and analysis of algorithms and data structures; asymptotic complexity, recurrence relations, lower bounds; algorithm design techniques such as incremental, divide-and-conquer, dynamic programming, iterative improvement, greedy algorithms; randomized data structures and algorithms. Prerequisites: COMP 2370 or equivalent and COMP 3200.

COMP 3372 Advanced Algorithms (4 Credits)
Advanced techniques for the design and analysis of algorithms and data structures; amortized complexity, self-adjusting data structures; randomized , online, and string algorithms; NP-completeness, approximation and exact exponential algorithms; flow networks.

COMP 3381 Software Engineering I (4 Credits)
An introduction to software engineering. Topics include software processes, requirements, design, development, validation and verification and project management. Cross-listed with COMP 4381. Prerequisites: COMP 3351; COMP 3361 or COMP 2362; or instructor permission.

COMP 3382 Software Engineering II (4 Credits)
Continuation of COMP 3381. Topics include component-based software engineering, model-driven architecture, and service-oriented architecture. Prerequisite: COMP 3381.

COMP 3411 Web Programming II (4 Credits)
In this course you will learn how to develop a full-stack web application that is capable of serving dynamic content from a database. Furthermore, you will learn the core design concepts and principles that will enable you to develop scalable and easy to maintain webapplications - a set of skills that will serve you well in both your personal and professional projects in the future. Prerequisite: COMP 3410.

COMP 3412 Web Projects: Web Development III (4 Credits)
In this course you will learn how to develop, as a group, a full-stack web application that is capable of serving dynamic content from a database. We will use the MongoDB, ExpressJS, Angular, and Node.js (MEAN) software stack to work on a real-life problem presented to us by an external product owner. In the class we will use the Scrum framework for Agile development to work, as a software team, through several sprints of development. You will be peer reviewing each other throughout the course, and the product owner will also be reviewing your product through end-of-sprint demos as features are completed. The goal for this class is for it to be a fun, collaborative, and educational environment that demonstrates what it is like to work as a real software team. Prerequisite: COMP 3411.

COMP 3421 Database Organization & Management I (4 Credits)
An introductory class in database management systems covering both relational and non-relational databases with an emphasis on relational. Topics include database design, ER modeling, relational algebra, SQL, scripting, and embedded SQL. Each student will design, load, query and update a nontrivial database using a relational database management system (RDBMS). In addition, an introduction to a NoSQL database will be included. Graduate students will read one or two relevant technical papers and write a summary report. Prerequisites: for undergraduates: COMP 1353 or COMP 2673; for graduates: COMP 3005 or COMP3006 (MS Data Science).

COMP 3424 NoSQL Databases (4 Credits)
In this course, students learn what NoSQL databases are, learn to identify the differences between them, and gain a fundamental understanding between SQL, relational databases, and NoSQL databases. Students further explore which type of NoSQL database is the correct one given a use-cases, examining types, methods of communicating with it, contrasts to other NoSQL databases, performance and scalability. Prerequisites: for undergraduates, COMP 2355 or COMP 2361; for graduates: COMP 3005.

COMP 3431 Data Mining (4 Credits)
Data Mining is the process of extracting useful information implicitly hidden in large databases. Various techniques from statistics and artificial intelligence are used here to discover hidden patterns in massive collections of data. This course is an introduction to these techniques and their underlying mathematical principles. Topics covered include: basic data analysis, frequent pattern mining, clustering, classification, and model assessment. Prerequisites: COMP 2370.

COMP 3621 Computer Networking (4 Credits)
An introduction to computer networks with an emphasis on Internet protocols. Topics include: internet design, application layer protocols such as SMTP and HTTP, session layer protocols including TCP and UDP, the internet protocol (IP), link layer technology such as Ethernet, and security issues related to networking. Programming experience of client/server architectures using sockets and TCP/UDP through projects is emphasized. Prerequisites: for undergraduates: (COMP 2355 or COMP 2361) and COMP 2370; for graduates COMP 3004 and COMP 3006. Cross listed with COMP 4621.

COMP 3681 Networking for Games (4 Credits)
Implementing the networking code for multiplayer games is a complex task that requires an understanding of performance, security, game design, and advanced programming concepts. In this course, students are introduced to the networking stack and how this is connected to the Internet, learn how to write protocols for games, and implement several large games using a game engine that demonstrate the kind of networking and protocols required by different genres of games. In addition, tools are introduced that help understand and debug networking code, simplify the creation of protocols, and make the development of networking code easier.

COMP 3701 Topics in Computer Graphics (4 Credits)

COMP 3702 Topics in Database (4 Credits)

COMP 3703 Topics-Artificial Intelligence (4 Credits)

COMP 3704 Advanced Topics: Systems (4 Credits)

COMP 3705 Topics in Computer Science (1-4 Credits)

COMP 3731 Computer Forensics (4 Credits)
Computer Forensics involves the examination of information contained in digital media with the aim of recovering and analyzing latent evidence. This course will provide students an understanding of the basic concepts in preservation, identification, extraction and validation of forensic evidence in a computer system. The course covers many systems level concepts such as disk partitions, file systems, system artifacts in multiple operating systems, file formats, email transfers, and network layers, among others. Students work extensively on raw images of memory and disks, and in the process, build components commonly seen as features of commercial forensics tools (e.g. file system carver, memory analyzer, file carver, and steganalysis). Prerequisites: COMP 3361; COMP 2355 or 2361 for undergraduates; COMP 3006 for graduates.

COMP 3801 Introduction Computer Graphics (4 Credits)
Fundamentals of 3D rendering including the mathematics behind coordinate systems, projections, clipping, hidden surface removal, shadows, lighting models, shading models, and mapping techniques. Significant use of 3D APIs through shader programming is covered along with the basics of 3D model representation and animations. Satisfies "Advanced Programming" requirements for graduate students. Prerequisites: COMP 2370, MATH 1952 or 1962.

COMP 3820 Introduction to Game Programming (4 Credits)
Learn the fundamentals of game programming by creating 2D and 3D games using a modern game engine. Topics include working with 2D/3D art assets, character controllers, physics, camera systems, 2D and 3D animation integration, lighting, audio, user interfaces, artificial intelligence, and level design. The emphasis in this course is on game programming and implementing game mechanics in fully-working games. Prerequisites: COMP 2381 and COMP 2821.

COMP 3821 Game Programming I (4 Credits)
A continuation of Introduction to Game Programming, this course introduces advanced topics that are essential as future game programmers. Students have the opportunity to learn game engine architecture, 2D and 3D linear algebra for graphics, sprites, 2D and 3D animations, input handling, finite state machines, particle systems, user interfaces, game audio, and artificial intelligence for games. Prerequisites: COMP 2370 and COMP 3820.

COMP 3822 Game Programming II (4 Credits)
In this course, students learn how to work with a 3D game engine and build 3D games. Topics include algorithms, mathematics for 3D game engines, scene management, animations, 3D shaders, particle systems, physics for games, UIs, terrain systems, and working with higher-level scripting languages on top of the low-level implementation language. Prerequisites: COMP 3821. Suggested corequisite or prerequisite: COMP 3801.

COMP 3904 Internship/Co-Op in Computing (0-10 Credits)
Experiential learning through employment with a company to work in computer science. Students are expected to find the internship/co-op and connect their hiring manager with an advisor in the department to get approval. Requirements for approval include a job description that shows what will be accomplished throughout the quarter and a final report from the hiring manager to report on their performance so that a grade may be assigned. Prerequisites: COMP 2370 and approval of department. MS Data Science students: practical experience working under the supervision of data science employer and a data science faculty advisor. Learning outcomes will be assessed via student work such as summary reports, literature reviews, presentations, code notebooks and/or repositories. Prerequisites: permission of Data Science Faculty Director.

COMP 3991 Independent Study (1-10 Credits)
Cannot be arranged for any course that appears in the regular course schedule for that particular year.

COMP 4100 Human-Computer Interaction (4 Credits)
Introduces students in computer science and other disciplines to principles of and research methods in human-computer interaction (HCI). HCI is an interdisciplinary area concerned with the study of interaction between humans and interactive computing systems. Research in HCI looks at cognitive and social phenomena surrounding human use of computers with the goal of understanding their impact and creating guidelines for the design and evaluation of software, interfaces, physical products, and services in industry. No prerequisites are required to take the course and students from all disciplines are welcome. Cross listed with COMP 3100.

COMP 4333 Parallel and Distributed Computing (4 Credits)
Current techniques for effective use of parallel processing and large scale distributed systems. Programming assignments will give students experience in the use of these techniques. Specific topics will vary from year to year to incorporate recent developments. This course qualifies for the Computer Science "Advanced Programming" requirement. Prerequisites: COMP2370 and COMP2355, or equivalent.

COMP 4334 Parallel and Distributed Computing for Data Science and AI (4 Credits)
Current techniques for effective use of parallel processing and large-scale distributed systems for data science. Programming assignments will give students experience in the use of these techniques. Specific topics will vary from year to year to incorporate recent developments. This course is not to be used for the MS Computer Science. Graduate students are required to contribute at least one course assignment to their personal Github portfolio. Prior completion or concurrent enrollment in COMP 4432 is strongly recommended. Prerequisite: COMP 3006.

COMP 4355 Advanced System Programming (4 Credits)
This course covers programming in a UNIX environment, including use of common command line utilities, scripting, source control via Git, and integration of POSIX system calls into C/C++ code. These features will be leveraged to solve practical problems cleanly and efficiently. More emphasis will be placed on using these features than on how those features work. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4370 Algorthmic Problem Solving (4 Credits)
The course is intended for students who are familiar with programming syntax but have not had much experience writing computer programs to solve a problem stated as a high-level description. The course will run through multiple such problem descriptions, discuss the design of programs to solve those problems using popular data structures, and have students implement those designs using a programming language. This course does not count for MS Computer Science requirements. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4372 Advanced Algorithms (4 Credits)
Advanced techniques for the design and analysis of algorithms and data structures; amortized complexity, self-adjusting data structures; randomized , online, and string algorithms; NP-completeness, approximation and exact exponential algorithms; flow networks. Prerequisite: COMP 3371. Cross listed with COMP 3372.

COMP 4384 Secure Software Engineering (4 Credits)
This course is concerned with systematic approaches for the design and implementation of secure software. While topics such as cryptography, networking, network protocols and large scale software development are touched upon, this is not a course on those topics. Instead, this course is on identification of potential threats and vulnerabilities early in the design cycle. The emphasis in this course is on methodologies and paradigms for identifying and avoiding security vulnerabilities, formally establishing the absence of vulnerabilities, and ways to avoid security holes in new software. There are programming assignments designed to make students practice and experience secure software design and development. Prerequisites: COMP 3006, COMP 3361, COMP 3356.

COMP 4401 Introduction to Python for Data Science (2 Credits)
This eight-week course enables aspiring graduate students in data science to meet their Python programming proficiency prior to matriculation. Students who complete this course will be able to write and modularize code, design efficient control structures, perform basic data operations, debug errors, distinguish data types and basic structures, and work within an integrated development environment.

COMP 4431 Data Mining (4 Credits)
Data Mining is the process of extracting useful information implicitly hidden in large databases. Various techniques from statistics and artificial intelligence are used here to discover hidden patterns in massive collections of data. This course is an introduction to these techniques and their underlying mathematical principles. Topics covered include: basic data analysis, frequent pattern mining, clustering, classification, and model assessment. Prerequisites: COMP 4441 and COMP 4581.

COMP 4432 Machine Learning (4 Credits)
This course will give an overview of traditional machine learning techniques, their strengths and weaknesses, and the problems they are designed to solve. This will include the broad differences between supervised/unsupervised and reinforcement learning as well as associated learning problems such as classification and regression. Techniques covered, at the discretion of the instructor, may include approaches such as linear and logistic regression, support vector machines, kNN, decision trees, random forests, Naive Bayes, EM, k-Means, and PCA. After course completion, students will have a working knowledge of these approaches and experience applying them to learning problems. Graduate students will contribute and assignment or project to a digital portfolio on Github. Prerequisites: MATH 3009 and COMP 3006.

COMP 4433 Data Visualization (4 Credits)
This course explores visualization techniques and theory. The course covers how to use visualization tools to effectively present data as part of quantitative statements within a publication/report and as an interactive system. Both design principles (color, layout, scale, and psychology of vision) as well as technical visualization tools/languages will be covered. Graduate students will contribute one assignment or project to a personal portfolio on Github. Prerequisite: COMP 3006.

COMP 4441 Introduction to Probability and Statistics for Data Science & AI (4 Credits)
The course introduces fundamentals of probability for data science. Students survey data visualization methods and summary statistics, develop models for data, and apply statistical techniques to assess the validity of the models. The techniques will include parametric and nonparametric methods for parameter estimation and hypothesis testing for a single sample mean and two sample means, for proportions, and for simple linear regression. Students will acquire sound theoretical footing for the methods where practical, and will apply them to real-world data, primarily using R. Graduate students will contribute a course assignment or project to their personal portfolio in Github. Prerequisite: COMP 3009.

COMP 4442 Advanced Probability and Statistics for Data Science and AI (4 Credits)
This course builds on material in Probability and Statistics 1. Students will carry out model fitting and diagnostics for multiple regression, ANOVA, ANCOVA, and generalized linear models. Dimension reductions techniques such as PCA and Lasso are introduced, as are techniques for handling dependent data. The course introduces the principles of resampling and Bayesian Analysis. Students will acquire sound theoretical footing for the methods where practical, and will apply them to real-world data, primarily using R. Enforced Prerequisites: COMP 4441.

COMP 4447 Data Science Tools 1 (4 Credits)
Organizations are using data science to extract actionable insight from data. To highlight the hidden patterns in the data, this course equips students with essential sills for data collection, cleanup, transformation, feature engineering, summarization, visualization, and collaboration. Students will do assignments and a final project. This is a hands-on course. Students will use Python libraries, Linux commands, API requests, web-scraping, GitHub and various data sets to perform these activities. Prerequisites: COMP 3006.

COMP 4448 Data Science Tools 2 (4 Credits)
Building a successful predictive model is a multi-faceted process. This course focuses on hypothesis testing and the development of predictive models. Students will learn the data science process for model development while learning to program a variety of predictive algorithms. Students will do assignments and a final project. This is a hands-on course. Students will use Python libraries, Linux commands, and various data sets to perform these activities. Prerequisite: COMP 4447.

COMP 4449 Data Science & AI Capstone (4 Credits)
Data Science Capstone provides students an opportunity to demonstrate their expertise as data scientists. Students are expected to integrate prior knowledge and skills to design, develop, test, and present 'full-cycle' data science products, and apply them in real-world contexts. This includes assessing and communicating their value to decision-making. Two sets of challenges of increasing complexity are presented as mid-term and term projects. These two projects are implemented, documented, tested, and presented by the students or students' two-person teams. Prerequisites: COMP 4432.

COMP 4450 Machine Learning Operations (4 Credits)
This course introduces the engineering and deployment of machine learning systems. Students will learn essential technical and organizational skills needed to develop and deploy ML solutions, progressing from fundamental concepts through to practical implementation. Key topics include ML project lifecycle management, containerization, web application development, data management strategies, and cloud deployment. Focus will be given to the application of industry-standard tools and best practices for building production-ready machine learning systems. Prerequisites: COMP 3421 and COMP 4432.

COMP 4455 Shell Scripting and System Tools (4 Credits)
This course covers navigating and utilizing tools in a UNIX environment, including use of common command line utilities, Bash and Python shell scripting, source control via Git, pipes and I/O redirection, networking in Python and OS multi-processing/multi-threading. More emphasis will be placed on using these tools than on how those tools work. Prerequisite: COMP 3006.

COMP 4456 Deep Learning for Sequence Data (4 Credits)
This course introduces modern approaches to analyzing sequential data, with a focus on deep learning methods for time series and natural language processing. Students will develop practical skills in building and implementing neural network models for sequential data analysis, while gaining a thorough understanding of the underlying concepts. Through hands-on projects and real-world applications, students will learn to process, analyze, and generate predictions from various types of sequential data. The course emphasizes practical implementation and modern deep learning frameworks, bridging the gap between traditional statistical methods and contemporary neural network approaches. Prerequisite: COMP 4432.

COMP 4490 Understanding AI (4 Credits)
Understanding Artificial Intelligence provides a comprehensive introduction to artificial intelligence (AI) for students of all backgrounds. This course explores the diverse approaches to AI, including rule-based systems, expert systems, knowledge representation, search algorithms, machine learning, natural language processing, robotics, and human-AI interaction. Students will gain hands-on experience with AI tools and applications while critically examining their impact on society. The course emphasizes the capabilities, limitations, and ethical considerations of AI, preparing students to engage thoughtfully with AI technologies in various fields.

COMP 4491 Applications of Generative AI (4 Credits)
Applications of Generative Artificial Intelligence is a hands-on course explores the transformative power of generative artificial intelligence (AI) across various industries. Students will engage directly with generative AI tools and techniques, including text generation, image synthesis, music composition, code generation, and more. Through practical exercises, projects, and case studies, students will develop a deep understanding of how generative AI can enhance creativity, automate workflows, and drive innovation in diverse fields. This course is designed for students who have completed COMP 4490 and want to apply AI in practical, industry-relevant contexts. Prerequisite: COMP 4490 and non-majors.

COMP 4501 Introduction to Artificial Intelligence (4 Credits)
Introduces a variety of Artificial Intelligence concepts and techniques, relevant to a broad range of applications. Students survey multiple techniques including search, knowledge representation and reasoning, probabilistic inference, machine learning, and natural language processing. Examines concepts of constraint programming, evolutionary computation and non-standard computation. Graduate students will also consider current political, economic, and societal issues related to artificial intelligence through reading and discussion. Prerequisites: COMP 3004 or equivalent.

COMP 4525 Adv. Human Centered AI - Mind Reading Machines (4 Credits)
This course explores a key intersection of Artificial Intelligence (AI) and Human-Computer Interaction (HCI). How can AI enhance our interactions with computers by catering to our individual needs and differences? Can we teach computers to understand people's thoughts and feelings to improve overall interaction? Humans often adjust their communication when they notice the person they're talking to is frustrated—why can't computers do the same? In this interdisciplinary, research-focused course, you will read, present, and discuss seminal papers at the intersection of AI and HCI, including (but not limited to) affective computing, physiological computing, augmented cognition, and multimodal interaction. ## Learning Objectives (General Goal) Demonstrate a broad understanding of the intersections between Human-Computer Interaction and Artificial Intelligence, with an emphasis on user modeling. (LO1) Recognize how theories from fields outside of Computer Science can inform and enhance Computer Science practices. (LO2) Develop knowledge of state-of-the-art user modeling techniques, including practical applications and associated challenges. (LO3) Develop critical thinking and scientific communication skills related to current AI topics, including science writing and scientific discourse. (LO4 - Graduate Students Only) Consider how the intersection of Humans and Artificial Intelligence impacts your own research and work. Consider the ethical challenges of your own work more deeply and how these may be mitigated.

COMP 4531 Deep Learning: Model Design and Application (4 Credits)
This course addresses the foundational concepts and components of Artificial Neural Networks (ANN), highlighting their capabilities, strengths, and weaknesses as a machine learning algorithm. Students taking this course will develop ANN models from scratch in Python as a basis for understanding their design as well as the underlying mechanics and calculations that shape their behavior. Key topics such as forward-backward propagation, loss function characteristics and optimization will be considered in relation to model design and computational efficiency as well as to problems such as exploding and vanishing gradients. Training strategies (e.g., dropout, initialization, batch normalization) will further enable students to assess trade-offs in model bias & variance. Coupled with hands-on assignments, these building blocks provide the knowledge and skills required to effectively design and implement ANN models that are ethically and technically sound. As well as foreground important architectures such as Convolutional ANNs, Recurrent ANNs, LSTMS, and Transformers as well as their applicability to modern problems. Student learning and proficiency will be assessed based on a combination of quizzes, coding assignments, exams, and a culminating project. Prerequisite: COMP 4432.

COMP 4581 Algorithms for Data Science & Artificial Intelligence (4 Credits)
This course introduces the design and analysis of algorithms within the context of data science and artificial intelligence. Topics include; asymptotic complexity and algorithm design techniques such as incremental, divide and conquer, dynamic programming, randomization, greedy algorithms, and advanced sorting techniques. Examples to illustrate techniques are drawn from multi-dimensional clustering (k-means and probabilistic), regression, decision trees, order statistics, data mining using apriori algorithms, and algorithms for generating combinatorial objects. Graduate students will contribute at least 1 assignment to their personal portfolio on Github. Prerequisite: COMP 3006.

COMP 4591 Computational Geometry (4 Credits)
This class deals with the design and implementation of efficient algorithms for problems defined over geometric objects, such as points, lines, polygons, surfaces, etc. The methods and algorithms covered find applications in many areas, including computer graphics (e.g., hidden surface removal), computer-aided design and manufacturing (e.g., 3D printing), machine learning (e.g., supervised and unsupervised classification), geographic information systems (e.g. terrain visibility), robotics (e.g., motion planning), data mining (e.g., dimensionality reduction), and computer vision (3D reconstruction), to name a few. Fundamental geometric problems such as partitioning, proximity, intersection, convexity, visibility, point location, and motion planning are focused on. Efficient data structures and algorithms for their solutions and design techniques germane to the field, such as divide-and-conquer, plane sweep, randomization, duality, etc. are discussed in detail. Practical methods for the robust implementation of geometric algorithms are also covered. Prerequisites: COMP 3200 and COMP 3371. This course satisfies the Theory requirement for graduate students.

COMP 4600 Seminar in Computer Science (0-4 Credits)
Preparation and presentation of lectures on some aspect of current research in computer science; topics not generally encountered in formal courses, may include robotics, pattern recognition, parallel processing, computer applications. 10- to 15- page paper with bibliography required.

COMP 4621 Computer Networking (4 Credits)
The Internet is arguably the most transformative invention in recent history and is at its core a massive global computer network (of networks). Students in this course learn how the Internet works, from the highest-level application layer to the lowest-level hardware layer. Topics covered include the OSI and TCP/IP reference models, physical transmission methods, error detection and correction, addressing, routing algorithms, congestion control and more. Prerequisites: COMP 3006, COMP 3361 (or instructor approval). Cross listed with COMP 3621.

COMP 4701 Special Tpcs-Computer Graphics (1-4 Credits)

COMP 4702 Advanced Topics-Database (3 Credits)

COMP 4703 Adv Topics-Artificial Intell (1-4 Credits)

COMP 4704 Advanced Topics-Systems (3-4 Credits)

COMP 4705 Advanced Topics-Programming (1-4 Credits)

COMP 4709 Special Tpcs-Computer Security (1-4 Credits)

COMP 4711 Special Topics in Data Science and AI (4 Credits)
Special Topics in Data Science highlights selected methods & applications in machine learning that are not specifically addressed within the curriculum. Areas of study may include, but are not limited to, natural language processing, recommender systems, data science for social justice, machine learning operations, generative AI and more. Special Topics in Data Science will be offered as an elective on an intermittent basis. Prerequisites: COMP 4432 and COMP 4441.

COMP 4721 Computer Security (4 Credits)
This course gives students an overview of computer and system security along with some cryptography. Some network security concepts are also included. Other concepts include coverage of risks and vulnerabilities, policy formation, controls and protection methods, role-based access controls, database security, authentication technologies, host-based and network-based security issues. Prerequisites: COMP 3006, COMP 3361 (or advisor/instructor approval).

COMP 4722 Network Security (4 Credits)
Network Security covers tools and techniques employed to protect data during transmission. It spans a broad range of topics including authentication systems, cryptography, key distribution, firewalls, secure protocols and standards, and overlaps with system security concepts as well. This course will provide an introduction to these topics, and supplement them with hands-on experience. In addition, students will perform an extensive analysis, or development of a security related product independently. Prerequisites: COMP 4721.

COMP 4723 Ethical Hacking (4 Credits)
Ethical hacking is the process of probing computer systems for vulnerabilities and exposing their presence through proof-of-concept attacks. The results of such probes are then utilized in making the system more secure. This course will cover the basics of vulnerability research, foot printing targets, discovering systems and configurations on a network, sniffing protocols, firewall hacking, password attacks, privilege escalation, rootkits, social engineering attacks, web attacks, and wireless attacks, among others. Prerequisites: COMP 3361, or COMP 3001, 3002, 3003, and 3004.

COMP 4724 Systems Security Management (4 Credits)
This course covers basic system administration tasks on a Unix environment, with a special focus on command line navigation, file/process access control, setting up network configurations, and managing services related to networks and their security. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4799 Capstone Project in Cybersecurity (1-8 Credits)
The purpose of the cybersecurity capstone project is to provide an integrative experience that ties together the learning outcomes from academic coursework undertakings and industry skills necessary to be productive in delivering an end product. Students will engage in one of many options available, such as involvement in a research project, a case study, a product development project, or an extensive survey paper. Capstone projects are presented at the end of the quarter in front of a representative group. Prerequisites: COMP 3001, 3002, 3003, and 3004.

COMP 4991 Independent Study (1-12 Credits)
Independent study on a particular topic supervised by a faculty member. Student must find a faculty member who will supervise work on the topic. Cannot be arranged for any course that appears in regular course schedule for that particular year.

COMP 4995 Independent Research (1-17 Credits)
Research projects undertaken in conjunction with a faculty member.

COMP 5991 Independent Study (1-17 Credits)

COMP 5995 Independent Research (1-17 Credits)