COURSE REFERENCE NUMBER: 23744 MWF 2:00pm–3:40pm from 06/10/2022 to 08/02/2022, online, via Zoom, as integrated in the Blackboard Learn system.

CREDIT HOURS: 3 hours.

COURSE DESCRIPTION: Introduction to Symbolic Computation via the computer algebra system SageMath. The mathematical algorithms in symbolic computation are explained with examples and applications to topics in undergraduate mathematics.

COURSE GOALS and LEARNING OBJECTIVES: The main goal of the course is to learn to apply SageMath to solve mathematical problems. The specific learning objectives are
(1) understand concepts of symbolic computation,
(2) gain *hands on* experience with computer algebra,
(3) learn mathematics through computations (computational thinking).
Extensive computer use is required. This is a *computational* course, not a programming class.

PREREQUISITES: MATH 210 (calculus III) and MCS 260 (introduction to computer science) or CS 107 (introduction to computing and programming); or CS 109 (programming for engineers with MatLab); or CS 111 (program design I); or consent of the instructor.

CURRICULUM: MCS 320 is the first course on the computational track, followed by MCS 471 (Numerical Analysis) and MCS 472 (Introduction to Industrial Math & Computation).

INSTRUCTOR and OFFICE HOURS: Jan Verschelde, Office: 1210 SEO, Phone: 312 996 4609.
Email: `janv@uic.edu`. URL: `http://www.math.uic.edu/~jan`.
Office hours are from 4PM to 5PM on Monday, Wednesday, and Friday, or by appointment; online, via zoom at `https://uic.zoom.us/my/profjanofficehour`.

TA GRADER: Ling Yu. Email `lyu37@uic.edu`. Office hour: on Tuesdays at 1pm via zoom at `https://uic.zoom.us/j/89441130822?pwd=kalq3NriqpW4JJCyQ9Y5Hl2bHTxI9b.1`

NO TEXT BOOK: There is no textbook for this course. Notes will be made available electronically via the course web site. A good reference is '*Sage for Undergraduates*' by Gregory V. Bard, AMS 2015 (ISBN 978-1-4704-1111-4); and available at `www.gregory-bard.com/Sage.html`.

SageMath AND THE JUPYTER NOTEBOOK: The free open-source mathematics software system `SageMath` is available online via `https://cocalc.com`. To install, visit `https://www.sagemath.org`. Answers to all homework, quizzes, projects, and exams must be submitted via gradescope in a Jupyter notebook with a SageMath kernel.

MCS 320 SITE: See `http://www.math.uic.edu/~jan/mcs320` for syllabus, notebooks, project descriptions, links to resources, etc... Backup site: `https://janv.people.uic.edu/mcs320`.

HOMEWORK: Exercises are assigned with each lecture. Although only a selection of the homework can be collected to make up for quizzes, it is strongly recommended to try all assignments.

QUIZZES: There will be a quiz every Wednesday, except during exam weeks. Every quiz is worth 20 points. There will be no makeup quizzes. If you miss a quiz or if your performance is bad, you can turn in the assigned homework to regain some of the points lost.

PROJECTS: Three projects will be assigned during the semester, worth jointly a total of 200 points.

EXAMS: During the semester, there will be two exams worth 100 points each. There will be no makeup exams given. The final exam counts for 200 points. If an exam is missed, then greater weight will be placed on the final exam, especially on the material covered on the missing exam.

GRADING SCALE: 90 – 100% = A, 80 – 89% = B, 70 – 79% = C, 60 – 69% = D, 0 – 59% = F.
Your course grade is based on a total of 700 points: 100 from the quizzes, 200 from the projects, 200 from the exams during the semester, and 200 from the final exam.

ACADEMIC HONESTY: No student shall claim or submit the work of another as ones own. Allowing someone to copy from you is also punishable. By default, unless stated otherwise, all your work in this class will be individual. If permitted to work in teams, then you must contribute your fair share.

Verbatim copying the output of generative AI tools is plagiarism.

POLICY FOR MISSED OR LATE WORK: Deadlines may be postponed. If you know you will be late, then it is better to apply for an extension of the deadline, instead of not submitting anything. For missed assignments, greater weight may be placed on the final project and/or final exam.

STUDENTS WITH DISABILITIES: UIC is committed to full inclusion and participation of people with disabilities in all aspects of university life. Students who face or anticipate disability-related barriers while at UIC should connect with the Disability Resource Center (DRC) at drc.uic.edu, drc@uic.edu, or at (312) 413-2183 to create a plan for reasonable accommodations. In order to receive accommodations, students must disclose disability to the DRC, complete an interactive registration process with the DRC, and provide their course instructor with a Letter of Accommodation (LOA). Course instructors in receipt of an LOA will work with the student and the DRC to implement approved accommodations.

CLASS ATTENDANCE: Students are expected to attend all class meetings. Any changes in this syllabus or in the scheduling of exams and other assignments will be announced during class meetings. We will also address the topics you need to implement the projects.

CLASSROOM CONDUCT POLICY: Respect others in the class. Students who engage in any behavior that results in the disruption of a class may be directed to leave the class room for the remainder of the class period.

SOME IMPORTANT DATES:

Friday 14 June : Last day to register, last day to withdraw without W grade.
Wednesday 19 June : Juneteenth holiday. No classes.
Thursday 4 July : Independence Day holiday. No classes.
Friday 12 July : Last day for optional late drop.
Friday 2 August : final exam.

LAST REVISED: Wednesday 5 June 2024.

COURSE OUTLINE – subject to changes :

Mon 10 Jun	L-1	welcome to MCS 320 — SageMath and CoCalc — computer algebra
	L-2	the Jupyter notebook — structuring and documenting your work
Wed 12 Jun	L-3	use as a calculator — getting started and getting help
	L-4	exact and floating-point numbers — continued fractions
Fri 14 Jun	L-5	complex and algebraic numbers — making finite fields
	L-6	symbols, variables, and references — preventing evaluation
Mon 17 Jun	L-7	number types — functions to store data
	L-8	evaluation and execution — finite fields — binary expression trees
Wed 19 Jun		**Juneteenth holiday. No classes.**
Fri 21 Jun	L-9	input/output formats — saving and loading data from files
	L-10	timing Python functions — vectorization — Cython
Mon 24 Jun	L-11	univariate and multivariate polynomials — coefficient types — monomial orders
	L-12	quotients of polynomials, expression swell, Horner form, partial fractions
Wed 26 Jun	L-13	operators and operands — expression trees for symbolic and numeric evaluation
	L-14	manipulating expressions by substitution, expansion, and factorization

Project One is due on Friday 28 June, at 2pm.

Fri 28 Jun	L-15	normalizing expressions — canonical forms
	L-16	review of the first 15 lectures
Mon 1 Jul	L-17	**first midterm exam on the first 15 lectures**
	L-18	second part of the first midterm exam
Wed 3 Jul	L-19	defining mathematical functions — preventing premature evaluation
	L-20	recursive functions and memoization for efficient execution
Thu 4 Jul		**Independence Day holiday. No classes.**
Fri 5 Jul	L-21	computing with functions
	L-22	symbolic, numeric, and implicit differentiation
Mon 8 Jul	L-23	integration and summation
	L-24	series, approximations, and limits
Wed 10 Jul	L-25	symbolic-numeric computation
	L-26	two dimensional plots
Fri 12 Jul	L-27	plotting in three dimensions and beyond
	L-28	making animations
Mon 15 Jul	L-29	solving equations
	L-30	linear algebra — matrix factorizations as normal forms

Project Two is due on Wednesday 17 July, at 2pm.

Wed 17 Jul	L-31	differential equations — Laplace transforms
	L-32	linear programming and polyhedra
Fri 19 Jul	L-33	building interactive web pages
	L-34	an application of interact
Mon 22 Jul	L-35	review of lectures 19 to 34
	L-36	more review for the second midterm
Wed 24 Jul	L-37	**the second midterm on lectures 19 to 34**
	L-38	second part of the second midterm exam
Fri 26 Jul	L-39	symbolic computing with sympy
	L-40	numeric computing with numpy and scipy
Mon 29 Jul	L-41	introduction to Julia
	L-42	parallel computing with Julia

Project Three is due on Wednesday 31 July, at 2pm.

Wed 31 Jul	L-43	review of the first half of the course
	L-44	review of the second half of the course
Fri 2 Aug	L-45	**final examination**
	L-46	second part of the final exam