Instructor
Dr. Kenneth W. Regan, 326 Davis Hall, 645-4738, regan@buffalo.edu;
Office hours: Tue. 1–2:30pm + TBA.

Lectures
(LEC) MWF 2:00pm–2:50pm in Norton 213

Reading—notes to be given in class, no textbook purchase
1. Notes by Arun Debray for Stanford’s undergraduate course as taught by Ryan Williams (who is now at MIT). They are based on the textbook by Sipser listed below (which is used even for graduate classes at MIT) but Williams’s priorities match my own very closely.

2. Chapters 27 and 28 of the CRC Handbook on Algorithms and Theory of Computing, co-authored by me with Professors Eric W. Allender and Michael C. Loui. These are for the second half of the course and will be given out in class.

3. Excerpts from my textbook with Richard Lipton, Quantum Algorithms Via Linear Algebra. These will also be given out.

4. Some optional components are yet to be determined. The weblog “Gödel’s Lost Letter and P=NP” may be used for assigned readings.

Optional Alternate Sources
1. Steven Homer and Alan Selman, Computability and Complexity Theory. The previous textbook.


3. J. Hopcroft and J. Ullman, Introduction to Languages, Automata Theory, and Computation, Addison-Wesley, 1979. The classic text. This course will mostly parallel the material in chapters 7–13 of this text; all assumed background and much more is in chapters 1–6.


5. N. Cutland, Computability, Cambridge University Press, 1980. A short-but-comprehensive and crystal-clear treatment of computability theory, the main topic of the first part of the course.
Examinations:
- Two prelim exams held in class period.
- One cumulative 3-hr. final.

Organization: The course will be graded on a total-points system. Letter grades will also be given for individual exams and possibly some assignments, as a help in telling you where you stand, but only the point totals will have official significance. The weighting of grades in this course shall be:

- Homework: 36%
- Prelims: 24%
- Final: 40%

I reserve the right to 5% leeway in weighting while assigning the final letter grade—this is most typically done for students who do markedly well on the final exam, when it may be treated as if it were worth 45% for that student. This will only be done to an individual student’s advantage, and will have no effect on others’ grades. The first prelim exam is tentatively set for Wed. Oct. 10.

The homework will consist of weekly or bi-weekly problem sets. All submissions will be in hardcopy. Some problems may be regarded as non-graded exercises. The first assignment will be given on Wed. Sept. 5.

Problem set submissions must be your own individual work. No joint submissions will be accepted. In an early lecture I will explain the purpose of individual work, academic integrity, and the “qualitative” nature of exercises in this course. I will give guidelines on how work can be done and what can be discussed among you. Cheating will be punished as per department policy at https://engineering.buffalo.edu/computer-science-engineering/information-for-faculty-and-staff/academic-integrity.html

My (KWR) general policy is not to implement a lateness-for-reduced-credit scheme. Instead I say that late work is not acceptable but extensions may be granted on request. Especially in smaller classes I am liberal with extensions, especially the 24-hour kind, but I still wish a request. In return, you get an answer key shortly afterward, and a relatively quick turnaround of graded work before the next problem set is due. In an exceptional situation, you may contact me beforehand.

Approximate Course Calendar (“the” syllabus)

The plan is to cover finite automata and (non-)regular languages in the first three weeks (plus a day), then computability and undecidability through mid-term. The second half will feature computational complexity (using my chapters with Allender and Loui and Debray’s notes as parallel texts): time and space complexity defined, why we emphasize \( P \) and \( NP \), \( NP \)-hardness and completeness, other salient complexity classes and the (known and unknown) relationships among them. Basics of randomized algorithms and quantum computing will round out the coverage. Homworks or Piazza posts will give indication from week to week of exactly what to read. I cannot spell out a timetable in greater detail now because my lectures will adjust to the needs of the class. I welcome feedback to me personally.

Reading Weeks 1 and 2: Debray’s notes up through the end of section 3 on page 11, in tandem with the slides notes called “Extra notes for Weeks 2–3” on the course webpage.