CS 521: Linear Programming
(Fall 2022)

Course Syllabus

1 Course Information

General information

Instructor: Sepehr Assadi. Email: `sepehr.assadi@rutgers.edu`. Office: CoRE 310.

Lectures: Fridays 10:20am – 1:20pm at PH-111 (Busch campus).

Office hours: Fridays 2:00pm – 3:00pm in CORE 310 or on Zoom (email the Instructor for the link).

Prerequisites: Undergraduate courses on algorithms, complexity theory, discrete mathematics, linear algebra, and probability; mathematical maturity. An advanced graduate course on algorithms (like CS 513 or 514) is strongly encouraged.

Textbook: There is no official textbook for this course but some useful resources are listed on the course webpage.

Webpage: https://sepehr.assadi.info/courses/cs521-f22/

The webpage contains updated syllabus information as the semester progresses and a calendar. The lecture notes and problem sets will also be posted on the webpage. We will use Canvas for the announcements and releasing grades.

Students with disabilities: The students with disabilities are encouraged to discuss with me any appropriate accommodations that we might make on their behalf following the guidelines of the Office of Disability Services[1].

Statement of inclusivity: I am committed to creating a learning environment in which all of my students feel safe and included, regardless of race, ethnicity, religion, gender or sexual orientation. Because we are individuals with varying needs, I rely on your feedback to achieve this goal. I invite you to let me know about what I can stop, start, or continue doing to make sure every one of my students feels valued and can engage actively in our learning community.

Academic integrity: The students are expected to follow Rutgers and CS Department academic integrity policies[2] for all their work in this course.

Mask requirements: In accordance with Rutger's policy, masks must be worn during class meetings; any student not wearing a mask will be asked to leave.

[1] https://ods.rutgers.edu

[2] Rutgers policy: http://academicintegrity.rutgers.edu/
CS department policy: https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy

What should you expect to learn from this course?

Linear programming (LP) is an extremely general and popular method of optimization in various domains including computer science, operations research, engineering, and economics. This course primarily focuses on the **theoretical foundations** of LPs including modeling concepts, theory, algorithms, and theoretical applications.

The following is a tentative list of topics (not necessarily in order) that will be covered in this course.

- Canonical LP Formulations
- Examples of LP: Shortest Path, Maximum Matching, Maximum/Minimum Cost Flows, ...
- Cutting Plane Methods, Center of Gravity
- Convexity and Basics of Convex Optimization: Gradient Descent
- Special Cases: Covering and Packing LPs, Low-dimensional LPs, ...
- Fundamental Theorem of LP, Farkas's Lemma, ...
- Duality and Complementary Slackness
- The Simplex Algorithm
- Polynomial Time Algorithms: Ellipsoid and Interior-Point Methods
- Integer Programming, LP Relaxations, and Approximation Algorithms
- Basics of Matroid Theory and Combinatorial Optimization
- Advanced topics: LP Hierarchies, Extended Formulations, Smoothed Analysis, ...

Along the way, we will learn about various algorithmic, analytic, and structural results in the context of linear programming that are widely applicable to other areas of theoretical computer science as well.

The course is intended for computer science PhD students, as well as advanced graduate students from other disciplines, with the main purpose of equipping the students with required background to use linear programming as part of their own research toolkit.

2 Assignments and Grading

Grading

- 30% Problem sets
- 30% Final Exam
- 40% Scribe notes

Problem Sets

– **Timing:** There will be two or three problem sets in this course (depending on the size of the class) and a tentative schedule of release and due dates will be posted on the course calendar. Problem sets will be released on a Monday and are **due three weeks** later by **11:59 pm on Monday**.

– **Format:** Problem sets should be turned in on Canvas as **a single pdf file** containing the solutions in order. Moreover, **solutions must be typeset in LaTeX**. Simple instructions on using LaTeX are available on the course webpage and a template will be released with each problem set.

– **Collaboration:** Problem sets should be done in teams of three to four students. Each team is responsible for handing in *one* solution only – every member of the team should be familiar with the solution to every problem and they maybe asked to explain the written solution at random.

When writing your solutions, you are *allowed* to use materials not discussed in the class (say, related research papers) **as long as you cite these references appropriately**. However, you are *not* allowed to get help from someone who is not currently enrolled in the class.

– **Difficulty of the problem sets?** The problem sets are going to be considerably challenging. You have three weeks to attempt each problem set so plan your time accordingly. However, do not get discouraged if you cannot solve some of the problems as they can be really hard – the goal of these problems is to familiarize you with the type of technical problems that appear typically in research on algorithms, and, exactly as in research, you should not expect to be able to solve every problem.

Scribe notes

– For each lecture, there will be one team (the same as the problem sets) in charge of taking detailed notes, typing them in LaTeX, preparing any needed figures, and sending them to the Instructor.

– We will use a two phase approach for these notes:

(*i*) The first phase is a quick summary of the lecture in terms of bullet points on the topics covered no more than a page or two, without any detailed proofs or description (this can ideally be prepared during the lecture itself). The target audience of these notes is primarily the students who may have missed the lecture and need a quick glance on the topics covered that day (as well as a helpful resource for the scribes in preparing their detailed notes in phase (*ii*)).

This phase is due on **11:59pm on Friday** of the same day as the lecture.

(*ii*) The second phase is the detailed lecture notes going over all the algorithms and proofs described in the lecture in details with potential figures and extra materials provided by the scribes. The target audience of these notes is you – students – a couple of weeks after the lecture, when you already have forgotten what it was about. As such, every detail of the lecture should be addressed in the notes.

This phase is due on **11:59 on Thursday** before the next lecture.

These notes will be posted on the course website.

– The majority of your grades in this course are coming from these scribe notes. If the quality of the notes are not satisfactory (e.g., having incomplete or imprecise proofs, many typos, lack of proper illustrations and examples, etc.), you will be asked to revise the notes until we converge to a version that is useful for your classmates.

– The assignments of scribe notes will be fixed in a couple of weeks – the first couple of lectures are scribed by the Instructor as an example. A LaTeX template with proper instructions for scribing the notes will be released as well.

3 Rutgers CS Diversity and Inclusion Statement

Rutgers Computer Science Department is committed to creating a consciously anti-racist, inclusive community that welcomes diversity in various dimensions (e.g., race, national origin, gender, sexuality, disability status, class, or religious beliefs). We will not tolerate micro-aggressions and discrimination that creates a hostile atmosphere in the class and/or threatens the well-being of our students. We will continuously strive to create a safe learning environment that allows for the open exchange of ideas and cherished freedom of speech, while also ensuring equitable opportunities and respect for all of us. Our goal is to maintain an environment where students, staff, and faculty can contribute without the fear of ridicule or intolerant or offensive language.

If you witness or experience racism, discrimination micro-aggressions, or other offensive behavior, you are encouraged to bring it to the attention to the undergraduate program director and/or the department chair. You can also report it to the Bias Incident Reporting System[3].

4 COVID-19 Protocols

In order to protect the health and well-being of all members of the University community, masks must be worn by all persons on campus when in the presence of others (within six feet) and in buildings in non-private enclosed settings (e.g., common workspaces, workstations, meeting rooms, classrooms, etc.). Masks must be worn during class meetings; any student not wearing a mask will be asked to leave.

Masks should conform to CDC guidelines and should completely cover the nose and mouth[4].

If you are feeling sick, or suspect you may have been exposed to COVID-19, do not come to the class. Arrangements will be made for students who are not able to attend class because of an illness or quarantine.

[3] http://inclusion.rutgers.edu/report-bias-incident/
[4] https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/about-face-coverings.html