JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

Syllabus
Computer Science 600.470
Combinatorics and Graph Theory in Computer Science
Spring, 2015
(3 credits, EQ)

Description

This is a graduate level course studying the applications of combinatorics and graph theory in computer science. We will start with some basic combinatorial techniques such as counting and pigeon hole principle, and then move to advanced techniques such as the probabilistic method, spectral graph theory and additive combinatorics. We shall see their applications in various areas in computer science, such as proving lower bounds in computational models, randomized algorithms, coding theory and pseudorandomness.

Prerequisites

Discrete math. Probability theory and linear algebra highly recommended.

Instructor

Professor Xin Li, `lixints@cs.jhu.edu`, `www.cs.jhu.edu/~lixints`
Office: Malone Hall 215, 410-516-5847
Office hours: Tuesdays 2:30–3:30 pm or by appointment

Meetings

Tuesday and Thursday, 12:00–1:15 pm, Shaffer 304

Textbook

Required: Stasys Jukna, *Extremal Combinatorics: With Applications in Computer Science*. 2nd Edition.
Optional: N. Alon and J. H. Spencer, *The Probabilistic Method.*

Online Resources

Any related online material will be posted at the course website http://www.cs.jhu.edu/˜lixints/class/spring15.html.

Tentative List of Topics

- Basic Techniques: Counting; Pigeon hole principle and resolution refutation lower bound; Matching and Hall's theorem.
- The Probabilistic Method: Basic method; Lovaz local lemma and its constructive proof; Linearity of Expectation; The deletion method; The entropy function; Random walks and randomized algorithm for CNF formulas.
- Spectral Graph theory: Basic properties of graph spectrum; Cheeger's inequality and approximation of graph expansion; Expander graphs and applications to superconcentrators and pseudorandomness; Error correcting codes and expander codes; Small set expansion, Unique Games Conjecture and Hardness of approximation.
- Additive Combinatorics: Sum product theorem, Szemeredi-Trotter theorem, Kakeya set problem and applications to randomness extractors.

Grading

There will be four or five homework problem sets, one mid-term exam and one final exam/project. Grading will be based on the following rule:

- Homework: 40%.
- Mid-term exam: 30%.
- Final exam/project: 30%.

Key Dates

- The mid-term exam will be held in class on March 12. The final exam (if any) will be held 9am to 12pm, May 14. No make-up exams will be given, so plan accordingly. For the midterm, you may bring a single, 8.5x11 inch, handwritten sheet of paper (you may use both sides); for the final you may bring two such sheets. No calculators are allowed (they won't be necessary).
- Should we choose final project instead of final exam, the project will include a presentation and a report. Typical projects include presentation of a paper, solving a known problem using a new method, or even working on a partial research problem. Suggested papers will be given later in the semester.

Ethics

The strength of the university depends on academic and personal integrity. In this course, you must be honest and truthful, abiding by the *Computer Science Academic Integrity Policy*:

> Cheating is wrong. Cheating hurts our community by undermining academic integrity, creating mistrust, and fostering unfair competition. The university will punish cheaters with failure on an assignment, failure in a course, permanent transcript notation, suspension, and/or expulsion. Offenses may be reported to medical, law or other professional or graduate schools when a cheater applies.
>
> Violations can include cheating on exams, plagiarism, reuse of assignments without permission, improper use of the Internet and electronic devices, unauthorized collaboration, alteration of graded assignments, forgery and falsification, lying, facilitating academic dishonesty, and unfair competition. Ignorance of these rules is not an excuse.
>
> Academic honesty is required in all work you submit to be graded. Except where the instructor specifies group work, you must solve all homework and programming assignments without the help of others. For example, you must not look at anyone else.s solutions (including program code) to your homework problems. However, you may discuss assignment specifications (not solutions) with others to be sure you understand what is required by the assignment.
>
> If your instructor permits using fragments of source code from outside sources, such as your textbook or on-line resources, you must properly cite the source. Not citing it constitutes plagiarism. Similarly, your group projects must list everyone who participated.
>
> Falsifying program output or results is prohibited.
>
> Your instructor is free to override parts of this policy for particular assignments. To protect yourself: (1) Ask the instructor if you are not sure what is permissible. (2) Seek help from the instructor, TA or CAs, as you are always encouraged to do, rather than from other students. (3) Cite any questionable sources of help you may have received.
>
> On every exam, you will sign the following pledge: "I agree to complete this exam without unauthorized assistance from any person, materials or device. [Signed and dated]". Your course instructors will let you know where to find copies of old exams, if they are available.

[In addition, the specific ethics guidelines for this course are:

(1) *Collaboration policy:* While you should first think about homework problems on your own, I encourage you to discuss homework problems with your classmates. You don't lose points by having collaborators. However, you must write up your own solutions. Any two students found sharing the same paragraph in their homework will both receive 0 point for that problem, and risk further punishment (such as automatic failure of the class). Furthermore, you must acknowledge any collaboration by writing the names of your collaborators on the front page of the assignment.

(2) *Citation policy:* Try to solve the problems without reading any published literature or websites, besides the class text. If, however, you do use a solution or part of a solution that you found in the literature or on the web, you must cite it (otherwise it's considered cheating and you will receive 0 point for that problem). Furthermore, you must write up the solution in your own words. You will get at most half credit for solutions found in the literature or on the web.

(3) *Late Policy:* Homework are due at the beginning of the class (usually on Tuesday). For each assignment you have two late days. A day here means 24 hours (i.e., from 12pm to 12pm next day). Any time after 12pm the first day until 12pm the next day is counted as one day. If you hand in your homework one day late, you will receive 80% of the points you earn. If you hand in your homework two days late, you will receive 60% of the points you earn. Homework later than two days will not be graded.

Report any violations you witness to the instructor.
You can find more information about university misconduct policies on the web at these sites:

- Undergraduates: `e-catalog.jhu.edu/undergrad-students/student-life-policies/`
- Graduate students: `e-catalog.jhu.edu/grad-students/graduate-specific-policies/`

Students with Disabilities

Any student with a disability who may need accommodations in this class must obtain an accommodation letter from Student Disability Services, 385 Garland, (410) 516–4720, `studentdisabilityservices@jhu.edu`.