CSE 347 Analysis of Algorithms, Spring 2019

Monday and Wednesday, 4:00-5:30 PM, Hillman 60.
Instructor: Brendan Juba
Office Hours: 5:30-6:30, (after lecture) Hillman 60 -> Jolley 508. (I will answer questions in Hillman 60 following lecture, and return to Jolley 508 when there are no more questions.)
Email: bjuba@wustl.edu

Please use the class Piazza board for all course-related communication We will use it for course announcements. Email of a personal/confidential nature may be directed to the instructor at the email above.

Recitation Leaders

- Aaron Handleman, Section A
  Fri 1-2 PM, Cupples II L009
- Erik Tomasic, Section B
  Fri 2-3 PM, Cupples II L009
- Hengxuan Li, Section C
  Fri 3-4 PM, Cupples II L009

Teaching Assistants

- Zongyi Li (head TA)
  Office Hours: Sunday, 3-4pm
- Priyanshu Jain
  Office Hours: Sunday, 3-4pm
- Chauncey Hill
  Office Hours: Monday 3-4pm
- Adam Kern
  Office Hours: Tuesday 12:30-1:30pm
- Adrien Xie
  Office Hours: Wednesday noon-1pm
- Sylvia Sheng
  Office Hours: Thursday 12:30-1:30pm
- Lexie Sun
  Office Hours: Friday, 11am-noon
- Wentao Wu
  Office Hours: Saturday, 3-4pm

Course Information

- Course overview
- Policy on collaboration and academic integrity
- Guide to submitting homework electronically
- Piazza discussion board

Course Schedule

The schedule of topics for class meetings and of assignments may be subject to change as the semester progresses. Readings marked "KT" refer to sections of the course text relevant to each class. Assignments will be linked from the schedule as they are assigned.

<table>
<thead>
<tr>
<th>Lecture no.</th>
<th>Topic</th>
<th>Dates</th>
<th>Homework</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lec. 1</td>
<td>Divide-and-Conquer I Matrix multiplication and Strassen's algorithm similar to KT 5.5</td>
<td>1/14</td>
<td>Assigned: Review Ch. 2 of KT (asymptotic analysis). Review KT 5.1-5.2 (divide and conquer algorithms)</td>
</tr>
<tr>
<td>Lec. 2</td>
<td>Greedy Algorithms I Scheduling KT 4.1</td>
<td>1/16</td>
<td>Assigned: Homework 1</td>
</tr>
<tr>
<td>Holiday</td>
<td>MLK Day - no lecture</td>
<td>1/21</td>
<td></td>
</tr>
<tr>
<td>Lec. 3</td>
<td>Dynamic Programming I Weighted Scheduling KT 6.1-6.2</td>
<td>1/23</td>
<td>Assigned: Review Ch. 3 of KT (on graphs).</td>
</tr>
<tr>
<td>Lecture no.</td>
<td>Topic</td>
<td>Dates</td>
<td>Homework</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-------</td>
<td>----------</td>
</tr>
<tr>
<td>Lec. 4</td>
<td>Greedy Algorithms II Scheduling all requests; start Minimum Spanning Tree KT 4.5</td>
<td>1/28</td>
<td>Due: Homework 1 Assigned: Homework 2</td>
</tr>
<tr>
<td>Lec. 5</td>
<td>Greedy Algorithms III Two algorithms: Kruskal and Prim finish KT 4.5</td>
<td>1/30</td>
<td></td>
</tr>
<tr>
<td>Lec. 6</td>
<td>Divide-and-Conquer II: The closest pair problem KT 5.4</td>
<td>2/4</td>
<td>Due: Homework 2 Assigned: Homework 3</td>
</tr>
<tr>
<td>Lec. 7</td>
<td>Dynamic Programming II Knapsack and Sequence Alignment KT 6.4, 6.6</td>
<td>2/6</td>
<td></td>
</tr>
<tr>
<td>Lec. 8</td>
<td>Max Flow I Ford-Fulkerson Algorithm KT 7.1</td>
<td>2/11</td>
<td>Due: Homework 3 Assigned: Homework 4</td>
</tr>
<tr>
<td>Lec. 9</td>
<td>Max Flow II Min-Cut KT 7.2</td>
<td>2/13</td>
<td></td>
</tr>
<tr>
<td>Lec. 10</td>
<td>Reductions I Matchings and more KT 7.5, 7.10</td>
<td>2/18</td>
<td>Due: Homework 4</td>
</tr>
<tr>
<td>Exam 1</td>
<td>Midterm Exam 1 (in class)</td>
<td>2/20</td>
<td></td>
</tr>
<tr>
<td>Lec. 11</td>
<td>Reductions II Relating problems KT 8.1, start 8.2</td>
<td>2/25</td>
<td>Assigned: Homework 5</td>
</tr>
<tr>
<td>Lec. 12</td>
<td>Intractability and NP-completeness KT finish 8.2, 8.3, start 8.4</td>
<td>2/27</td>
<td></td>
</tr>
<tr>
<td>Lec. 13</td>
<td>Intractability via Reductions I KT finish 8.4, 8.8</td>
<td>3/4</td>
<td>Due: Homework 5 Assigned: Homework 6 Review basic probability, KT 13.12, KT 13.3</td>
</tr>
<tr>
<td>Lec. 14</td>
<td>Average-case analysis: Naive trees and hashing</td>
<td>3/6</td>
<td></td>
</tr>
<tr>
<td>Holiday</td>
<td>Spring Break - no lecture</td>
<td>3/11</td>
<td></td>
</tr>
<tr>
<td>Holiday</td>
<td>Spring Break - no lecture</td>
<td>3/13</td>
<td></td>
</tr>
<tr>
<td>Lec. 15</td>
<td>Randomized algorithms I From trees to treaps Roughly, KT 13.9; see also KT 13.9</td>
<td>3/18</td>
<td>Due: Homework 6 Assigned: Homework 7</td>
</tr>
<tr>
<td>Lec. 16</td>
<td>Intractability via Reductions II KT 8.5, 8.7</td>
<td>3/20</td>
<td></td>
</tr>
<tr>
<td>Lecture no.</td>
<td>Topic</td>
<td>Dates</td>
<td>Homework</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
<td>-------</td>
<td>----------</td>
</tr>
<tr>
<td>Lec. 17</td>
<td>Randomized algorithms II Hashing KT 13.6</td>
<td>3/25</td>
<td>Due: Homework 7 Assigned: Homework 8</td>
</tr>
<tr>
<td>Lec. 18</td>
<td>Randomized algorithms III: Algorithms that may err KT 13.2</td>
<td>3/27</td>
<td></td>
</tr>
<tr>
<td>Lec. 19</td>
<td>On-line algorithms I: Amortized Analysis KT 4.6</td>
<td>4/1</td>
<td>Due: Homework 8</td>
</tr>
<tr>
<td>Exam 2</td>
<td>Midterm Exam 2 (in class)</td>
<td>4/3</td>
<td></td>
</tr>
<tr>
<td>Lec. 20</td>
<td>On-line algorithms II: Competitive Analysis KT 13.8, 11.1</td>
<td>4/8</td>
<td>Assigned: Homework 9</td>
</tr>
<tr>
<td>Lec. 21</td>
<td>Approximation algorithms I Greedy and random KT 11.3, 13.4</td>
<td>4/10</td>
<td></td>
</tr>
<tr>
<td>Lec. 22</td>
<td>Approximation algorithms II Relaxing and rounding KT 11.6</td>
<td>4/15</td>
<td>Due: Homework 9 Assigned: Homework 10</td>
</tr>
<tr>
<td>Lec. 23</td>
<td>Approximation algorithms III Polynomial-time approximation schemes KT 11.8</td>
<td>4/17</td>
<td></td>
</tr>
<tr>
<td>Lec. 24</td>
<td>Smoothed Analysis</td>
<td>4/22</td>
<td>Due: Homework 10</td>
</tr>
<tr>
<td>Lec. 25</td>
<td>TBD</td>
<td>4/24</td>
<td></td>
</tr>
</tbody>
</table>
Overview of CSE 347 Analysis of Algorithms

Contents

- Course Goals and Content
- Interacting and Seeking Help
- Assignments and Grading

Course Goals and Content

This course introduces the mathematical analysis of algorithms. An algorithm is a procedure for solving a problem, expressed in a way that can be effectively implemented on a computer. Useful algorithms are both correct and efficient – they produce the right answer and run in a reasonable amount of time/space as a function of the size of the problem being solved.

We will study a number of different types of algorithms, such as divide-and-conquer, greedy, and dynamic programming. Our goals are to understand why these methods work and how to quantify their efficiency. "Efficiency" in computer science is traditionally measured as the worst-case time for an algorithm to process any one input of a given size. However, in some contexts, other measures of efficiency may be more useful, such as average-case efficiency, amortized efficiency over a large number of inputs, or competitive efficiency relative to some other algorithm. Also, clever use of random numbers in an algorithm may allow us to guarantee efficient execution "most of the time," regardless of the input. Finally, some problems are intractable; that is, they are inherently resistant to efficient implementation. We'll introduce the study of intractability and how to tell when a problem is likely intractable.

Because algorithms and their analysis can be technically complex, trying to understand their behavior based solely on informal arguments (sometimes called "hand-waving") is fraught with peril. The safest way to avoid being wrong is to pursue rigorous correctness and efficiency arguments (i.e. proofs). We will strive for mathematically rigorous argument, both in class and on your assignments. An important part of learning about algorithms is learning how to argue rigorously about their behavior; this class will give you lots of practice to help build up your "CS theory muscles."

After successfully completing this class, you should be able to understand and rigorously analyze a wide variety of algorithms encountered in practice. You should also be able to articulate what it means for a problem to be intractable, to recognize intractability in some cases, and to propose approaches to make progress on intractable problems. The skills you gain will be particularly important if you have occasion to design your own algorithms in the future; however, everyone in CS should be an "informed user" of algorithms.

Prerequisites

Formally, this class requires CSE 247. In particular, we expect that you are familiar with notions of asymptotic worst-case complexity and how to measure it, as well as basic data structures and fundamental methods for computational tasks such as sorting, maintaining dictionaries and search indices, and graph traversal. We also expect that you can make simple arguments in support of algorithmic correctness and complexity, at the level done in CSE 240 and 247.

Informally, 347 demands that you develop "mathematical maturity" -- the ability to read, understand, and produce rigorous mathematical arguments (i.e. proofs). We will use basic logical techniques such as induction, proof by contradiction, and proof by cases extensively, as well as elementary probability and combinatorics. You will practice communicating your ideas in the form of proofs through your classroom interactions, recitation sections, and homeworks.

Textbook

We will use Jon Kleinberg and Eva Tardos’s Algorithm Design. This is a required text -- some of the homework problems will be assigned from it. The course schedule indicates which sections of the text correspond to each class meeting.

Opportunities to Interact and Seek Help

Besides our regular class meetings, the instructors and TAs will hold weekly office hours. We will also offer weekly 60-minute recitation sections. In recitation, you will have the opportunity to work in small groups on solving computational problems and analyzing your solutions, and on communicating your analysis in the form of proofs. You will be required to register for one of the three recitation sections. We strongly recommend making time to regularly attend recitation, particularly if you have little prior experience with rigorous mathematical proof.

You can also interact with us through the class discussion board, which are hosted on Piazza. We will also use Piazza to maintain up-to-date information about office hours and to post class announcements. The Piazza board for this class is located at piazza.com/wustl/spring2019/cse347/home.

All students will be automatically signed up for Piazza by the start of class. If you have not yet registered on Piazza, you should receive an email inviting you to join. Please register as soon as you receive the e-mail, if you haven’t already.

You should use Piazza for all questions not of a personal or confidential nature. Your posts will be seen by the instructors, the other course staff, and (if you don’t make your post private) by your fellow students. You will find it
helpful when your classmates have already asked a question for you -- in fact, someone may already have answered it!
But, in order to sustain this, you must ask your questions to Piazza as well.

Because 347 is a large class, and your instructors and course staff already have very full email in-boxes, we will
strictly enforce a policy of responding only to class questions posted via Piazza. Please email your instructors
only for personal emails (e.g. letting us know about an illness or absence) or confidential matters (e.g. accusations of
collaboration policy violations). If you have a question about homework or class logistics that you don't want to share
with the class, please post it to Piazza, but make your post private.

Disability Resources

Students with documented disabilities are strongly encouraged both to bring any need for accommodation to the
attention of the instructors and to make full use of the University's disability resources and accommodations.

Assignments and Grading

Tentatively, there will be 10 weekly homework assignments, two midterm exams, and a cumulative final exam. You will
also be assigned credit for participating in activities during lecture and recitation. The course schedule contains a
tentative schedule for the homework assignments. Homeworks will be linked from the course schedule as they are
assigned.

Each homework assignment will contain three problems. We will use the top two scores out of the three problems to
assign a grade for your homework, i.e., we will drop the problem with the lowest score from your submission. Please
keep in mind that you will be responsible for the content of all of the homework problems on the exams, so we strongly
encourage you to attempt all of the problems we assign.

Electronic Submission of Homework

We will use Gradescope via Canvas for paperless homework submission. To allow for electronic submission, grading,
and enforcement of the collaboration policy, we require that you prepare your homework using a word processor or
typesetting program, rather than hand-writing it. (You can, however, hand-draw and scan in figures as part of your
solutions.) You can read about how to typeset your homework, including math and figures, and the rules and
procedures for submission, in our E-Homework Guide.

Typesetting your homework takes a little more time, at least initially, but you will find it helpful for problem solving. We
don't hand-write papers anymore because word processors offer huge advantages in terms of editing your work,
particularly rearranging, reusing, and replacing pieces over time. Think of your homework solutions as mini-essays
(which, in a certain sense, they are!), and you'll see why it might be helpful to write them electronically. Don't try to
hand-write a beautiful solution and then typeset it at the last minute -- use your word processor or editor to capture your
ideas and compose a solution as you go.

Deadlines and Late Homework Policy

Homework will be due at 11:59PM on its due date. If all problems for a given homework are not uploaded and submitted
through Gradescope by this deadline, the entire homework will be considered late. Homeworks turned in fewer than 24
hours after deadline are one day late; those turned in at least 24 but less than 48 hours after deadline are two days late;
and so forth.

The following policy applies to homeworks turned in late. Everyone is automatically granted two "late days" for each
assignment, for any reason. You are not required to contact course staff to use these late days. In other words, you will
be allowed to submit an assignment for credit up to two days past the date indicated on the schedule and on the top of
the homework assignment. But, we will not accept work submitted beyond these two days. Please note that every day
or partial day that a homework is late, including weekends and holidays, counts toward its total lateness.

Regrade requests

Please use a private message on Piazza to submit all regrade requests.

Collaboration and Academic Integrity Policy

Please see this page for the policy.

Homework Advice

The goal of homework is for you to practice the reasoning and analytical approaches that we develop in class to
understand new algorithms and design and analyze novel variations on the methods we've studied. Through these
homeworks, you will develop your problem-solving skills, learn to recognize which among many possible analytical
techniques can be applied, and enhance your ability to communicate clearly in the language of theoretical computer
science. Results proved in homework may sometimes be used later on in the course to support new material that we
discuss in class.

Algorithms homework can be challenging. Often, you may have to think about a problem and try alternative approaches
for a while before you can make progress. You might have to put the problem down and come back to it later. You might
intuitively see how to proceed but have difficulty articulating or justifying your approach formally. These are normal parts
of the problem-solving process, and you will get better at it over time.
Two things to keep in mind are that (1) we don't expect you to come up with all of the insights necessary to solve every problem by yourself, and (2) give yourself enough time to do the homework. Within the limits of the course collaboration policy, we encourage you to work on problems in groups, and to consult appropriate outside sources. If you get stuck, you can talk to an instructor or other course staffer or post to Piazza, so that we can try to steer you toward the next step in the solution.

In order to have sufficient time to work with your peers, seek help, and prepare your solutions, please start your homework early, and plan to devote significant time to it over the week. Leave yourself time to ask questions, to talk to others, and to put a problem down and come back to it. Don't assume that you can do the whole thing in one sitting.

**Breakdown of Final Grade**

The breakdown of your final grade will be as follows:

- participation - 10%
- homework - 20%
- midterm exam 1 - 20%
- midterm exam 2 - 20%
- final exam - 30%

Any curving of exams to adjust for observed difficulty is at the sole discretion of the instructors. We do not anticipate curving homework grades, and we will not drop exam scores.
CSE 347 Policy on Collaboration and Academic Integrity

This document outlines the course policy on academic integrity and collaboration on homeworks. You must read and understand this document at the beginning of the course. Submitting your homework through Gradescope indicates that you have read and agreed to comply with the policy. If you are uncertain as to how to comply with this policy, please ask an instructor.

Written Solutions Must be the Student's Own Words

You are permitted, and indeed encouraged, to discuss ideas, approaches, and so forth with other students, up to and including working out a solution together. You may also use outside sources, including books, websites, and other literature to arrive at a solution. However, the solutions that you submit must be your own, original words. You may not copy verbatim nor paraphrase text from another student or source.

We strongly recommend taking the following measures to avoid unintentionally violating this policy.

- Do not consult any written sources other than the textbook and official course materials while you are actually writing your solution.
- If you discuss a problem with other students, do not refer to written notes from that discussion when writing your solution.
- After consulting an outside source or discussing a problem with another student, wait at least one hour before writing any part of your own solution. This should prevent you from accidentally reproducing the language that someone else used to describe their approach.

Please note that we will run all submitted solutions through an automated process that checks for substantial similarity to other students' work and to published sources. This process has in the past caught violations of our policy.

Assistance Must Be Acknowledged

Each time you turn in a homework, you are required to list any sources of assistance you received in solving the problems. Please specify where you got help and what kind of help you received. For collaborations with other students, please list their names. For written sources, please provide a full reference -- author, title, publication date, URL, and so forth.

You may acknowledge assistance in your homework solutions themselves. Please record this information for each homework problem.

You need not acknowledge help received from the instructor or teaching assistants, or from the current semester's course materials or the textbook.

Sanctions

In cases where a student is found to have violated the course collaboration policy by submitting a solution with substantially the same text as another student's submission or some other source, the minimal penalty is loss of all credit for that solution (only). The penalty for impermissibly similar text applies even if the collaboration or source is acknowledged, and even if it occurred without conscious intent to deceive. If the source or collaborator is not acknowledged as part of the submission, more serious penalties will apply.

Depending on the severity of the violation, penalties may include partial or complete loss of credit for assignments, failure of the course, or such other disciplinary actions as are warranted and allowed by Washington University. Per CSE Department policy, incidents of known or suspected cheating will be reported to the School of Engineering (for undergrads and MS students) or the Graduate School (for PhD students) and may affect your ability to graduate.

The Spirit of the Policy

The goal of this policy is to promote collaboration and self-directed learning as part of the homework process, while still ensuring that the final product is reflective of each person's understanding. We've set some bright-line rules to promote this goal, but please consider more generally whether your homework practices are consistent with it.

If you are making progress on a problem, please don't just tell other students how to solve it or point them to a worked-out solution. Instead, try to communicate the principles by which they can move toward discovering their own solutions. If you ever feel uncomfortable discussing the homework, or you don't see how to help someone without simply giving away answers, please refer them to the instructors or other course staff rather than trying to sort things out on your own.

Keep in mind that your performance on exams, which are not collaborative, constitutes more than half your course grade, so it ultimately does you no good to let other people or sources "steal your struggle" by solving problems for you without developing your own skills. Conversely, you're not doing your fellow students a favor by handing them a solution. Collaboration should feel like a partnership -- even if one person has the first insight more often than another, everyone should contribute and should obtain lasting benefit from the experience.