Course Information

Instructors

Jason Ku	38-683	jasonku@mit.edu
Julian Shun	32-G736	jshun@mit.edu
Virginia Williams	32-G640	virgi@mit.edu

Teaching Assistants

Brynmor Chapman	BC	brynmor@mit.edu
Alex Chen	AC	achen19@mit.edu
Louisa Huang	LH	rlhuang@mit.edu
Shreyas Kapur	SK	shreyask@mit.edu
Ashley Kim	AK	ahkim@mit.edu
Sabrina Liu	SL	sabliu@mit.edu
Eric Qian	EQ	ericqian@mit.edu
Meena Rajan	MR	meenar@mit.edu
Martin Schneider	MS	martinfs@mit.edu
Aditi Srinivasan	AS	aditisri@mit.edu
Rishi Sundaresan	RS	rishisun@mit.edu
Erica Weng	EW	ericaw@mit.edu

Questions?

6.006-questions@mit.edu

Websites

Stellar	Announcements, calendar, grades, and PDF course content. https://tinyurl.com/6-006fa19
Piazza	All discussion related to course material. http://piazza.com/mit/fall2019/6006
Gradescope	LaTeX problem set submissions and regrades. Entry Code: MG6JD4 https://www.gradescope.com/courses/61235/
Code Checker	Auto-graded code problem set submissions. https://alg.mit.edu

Content

6.006 is an introductory course covering elementary data structures (dynamic arrays, heaps, balanced binary search trees, hash tables) and algorithmic approaches to solve classical problems (sorting, graph searching, dynamic programming). Written course material will be distributed via notes from lectures and recitations. An additional useful reference is **Introduction to Algorithms** by Cormen, Leiserson, Rivest, and Stein (Third Edition, MIT Press), commonly known as **CLRS**, though this text is not required for the course.

Prerequisites

6.0001	Basic experience programming in Python 3.
6.042	Basic knowledge of discrete mathematics: set theory, relations and logic, combinatorics, proofs, recursion, number theory, graph theory, and probability.

We **strongly** caution against taking 6.006 before having fulfilled the listed prerequisites. We will evaluate entering understanding of the prerequisite material via a short Problem Set 0 assignment (Released R 9/05 and Due on U 9/08). **All students must submit this evaluation, regardless of prerequisite status**. We will assign each submission a letter grade: A, B, or C. If you receive a C on the assignment, you will need to meet with a staff member to review your performance before you will be allowed to take the class. **We will not grade any other assignments from you until a good faith attempt of Problem Set 0 has been submitted**. The grade for this assignment will NOT effect your final grade in the class, but turning it in is required for taking this class.

Instruction

Lectures will be held in Room 26-100 from 11:05 A.M. to 11:55 A.M. on Tuesdays and Thursdays. One-hour **Recitations** will be held weekly on Wednesdays and Fridays, beginning on Wednesday, February 6th. Recitations supplement the material presented in lecture in a more interactive setting. You are responsible for material presented during both lecture and recitation. Initial recitation assignments have been provided by the registrar, though you may reassign yourself to any recitation section with available space via the Stellar website. Please finalize your recitation section choice by Friday, September 13th.

R01 @10 34-302 MR	R02 @10 34-303 EQ	R03 @11 34-302 EW	R04 @11 34-303 LH	R05 @12 34-301 AK	R06 @12 34-304 AC
R07 @1 34-304 AC	R08 @1 35-310 MS	R09 @2 35-310 AS	R10 @2 36-155 RS	R11 @3 36-155 SL	R12 @4 36-155 SK

Office Hours

Teaching Assistants will hold office hours every Sunday, Monday, Tuesday, Wednesday, and Thursday from 7-10 P.M. in Rm. 26-314 (with overflow next door in Rm. 26-322). Instructors will hold individual office hours by appointment.

Grading Policy

Your grade will be based on recitation, 9 problem sets, 3 quizzes, and a final exam.

	Weight	Date	Time	Location(s)
Quiz 1	20%	Thursday, October 10, 2019	7:30–9:30 P.M.	2-190, 6-120, 54-100
Quiz 2	15%	Thursday, November 7, 2019	7:30–9:00 P.M.	2-190, 6-120, 54-100
Quiz 3	10%	Thursday, December 5, 2019	7:30–8:30 P.M.	2-190, 6-120, 54-100
Final Exam	35%	Monday, December 16, 2019	9 A.M.–Noon	Johnson Track
Problem Sets	18%	9 Problem Sets, 2% each		
Recitation	2%	Graded by your TA		

MIT provides definitions[1] for the letter grades A, B, C, D, and F. In order to normalize assignments that vary in length and difficulty, we will provide a separate piece-wise linear mapping for each assignement, from the assignment's grade space to the interval $[0, 5]$. Grades mapped to the half-open intervals $(4, 5]$, $(3, 4]$, $(2, 3]$, $(1, 2]$, and $[0, 1]$ correspond to letter grades A, B, C, D, and F respectively. To determine your final grade in the class, we will compute the weighted sum of your mapped assignment grades. A sum strictly above 4 will be at least an A-, above 3 will be at least a B-, etc., but we reserve the right to lower course letter grade cutoffs at the end of term based on institute grade definitions.

If you feel that any assignment has been graded incorrectly, you may submit a **regrade request** to the relevant assignment on Gradescope, within a regrade window after the assignment's grade has been released (typically about a week). For any regrade request, we reserve the right to regrade the **entire assignment**, and your grade may be adjusted **up or down** as a result of the regrade.

Exams

There will be no official lecture on quiz days. A review will be given during the recitation preceding each quiz. Quizzes and the Final Exam will be closed book, but you will be allowed to bring and use some pre-prepared **double-sided notes**: one page for Quiz 1, two for Quiz 2, three for Quiz 3, and three for the Final Exam. **Attendance at the quizzes and the Final Exam is mandatory and may not be excused.** A quiz may be rescheduled at the emailed request of an Institute Dean, to `6.006-questions@mit.edu`. Course-wide makeup quizzes will be given within a day of the scheduled date. Conflict Final Exams will be scheduled by the registrar.

[1] `http://catalog.mit.edu/mit/procedures/academic-performance-grades/#gradestex`

Problem Sets

PS	Release	Due	Topic
0	R 9/05	U 9/08	Prerequisite Evaluation
1	F 9/06	F 9/13	Asymptotics, Sequences
2	F 9/13	F 9/20	Sets, Sorting, Recurrences
3	F 9/20	F 9/27	Hashing, Linear Sorting
4	F 9/27	F 10/04	Binary Trees, Binary Heaps
5	F 10/11	F 10/18	Graph Traversal
6	F 10/18	F 10/25	DAG Relaxation, Bellman-Ford
7	F 10/25	F 11/01	Dijkstra's, Johnson's
8	F 11/08	F 11/15	Dynamic Programming
9	F 11/15	F 11/22	More Dynamic Programming

Each problem set will contain a theory portion and a coding portion. Each theory portion must be uploaded to Gradescope as a PDF file compiled from a provided LaTeX template. Each coding portion will be administered and automatically graded via our Code Checking website, and must be completed using Python 3. Problem set submissions are **due by 6 P.M.** on the posted due date.

Late submissions will be accepted up until 48 hours after the due date, also at 6 P.M.. Solutions will be posted shortly after the late submission window closes. We will not penalize your two highest scoring late submissions, but we will penalize any additional late submissions by 50%. In exceptional circumstances, problem set deadlines may be individually extended without penalty at the emailed request of an Institute Dean, to `6.006-questions@mit.edu`. As every assignment contributes to learning, **no assignment may be dropped.**

Collaboration

The goal of the problem sets is for you to practice applying the course material. In this class, you are **encouraged** to collaborate on problem sets. Students who work together on problem sets generally do better on exams than students who work alone, but you will learn the material best if you **work on the problems FIRST on your own**. Some forms of collaboration are **not allowed**; some examples are listed below. Violating the collaboration policy to increase your score on a problem set is likely to lower your score on an exam, which carries significantly more weight. A violation may also lead to academic action and/or a significant penalty on your grade.

- Identify any **collaborators** or **outside sources** at the top of each LaTeX submission.
- Write code and theory problem solutions **by yourself** in your own words.
- Do **NOT** directly copy the work of others.
- Do **NOT** look at written solutions or code by other students before submitting your own solution. You may look at another student's code on their screen, only to help them debug, and only after you have submitted your own solution.
- Do **NOT** let other students see your written solutions.
- Do **NOT** send other students your code.
- You may ask TAs to help you debug your code during office hours or in a private Piazza post.

Syllabus

	Date	Lec	Topic		Date	Rec	Topic
R	9/05	L01	Algorithms and Computation	F	9/06	R01	Asymptotics
T	9/10	L02	Data Structures / Dynamic Arrays	W	9/11	R02	Python Lists
R	9/12	L03	Search / Sort	F	9/13	R03	Recurrences / Master Theorem
T	9/17	L04	Direct Access & Hashing	W	9/18	R04	Python Dicts and Sets
R	9/19	L05	Linear Sorting	F	9/20		Career Fair
T	9/24	L06	Balanced Binary Trees	W	9/25	R06	Binary Trees in Python
R	9/26	L07	BSTs and Sequence Trees	F	9/27	R07	BSTs in Python
T	10/01	L08	Heaps / Priority Queues	W	10/02	R08	Heaps in Python
R	10/03	L09	Breadth-First Search	F	10/04	R09	Graph Traversal in Python
T	10/08	L10	Depth-First Search	W	10/09	R10	Quiz 1 Review
R	10/10		**Quiz 1: L01 – L08**	F	10/11		
T	10/15		Columbus Day	W	10/16		
R	10/17	L11	Weighted Shortest Paths	F	10/18	R11	Relaxation
T	10/22	L12	Bellman-Ford	W	10/23	R12	Bellman-Ford in Python
R	10/24	L13	Dijkstra	F	10/25	R13	Dijkstra in Python
T	10/29	L14	All-Pairs Shortest Paths	W	10/30	R14	Johnson's Algorithm & Review
R	10/31	L15	Dynamic Programming Intro	F	11/01	R15	Dynamic Programming in Python
T	11/05	L16	Guessing Subproblems	W	11/06	R16	Quiz 2 Review
R	11/07		**Quiz 2: L01 – L14**	F	11/08		
T	11/12	L17	Subproblem Expansion	W	11/13	R17	Dynamic Programming Examples
R	11/14	L18	Dynamic Programming Shortest Paths	F	11/15	R18	Dynamic Programming Examples
T	11/19	L19	Subset Sum & Pseudo-polynomial	W	11/20	R19	Partition & 0-1 Knapsack
R	11/21	L20	P, NP, Hardness, Completeness	F	11/22	R20	Complexity
T	11/26	L21	Specialized Algorithms	W	11/27		
R	11/28		Thanksgiving	F	11/29		
T	12/03	L22	Algorithms Research	W	12/04	R22	Quiz 3 Review
R	12/05		**Quiz 3: L01 – L20**	W	12/06		
T	12/10	L23	Course Review	W	12/11	R23	Final Review
M	12/16		**Final: L01 – L23**				