



# CS165

## Data Systems

Have fun learning to design and build modern data systems

## CS165 FALL 2023 SYLLABUS

[HTTP://DASLAB.SEAS.HARVARD.EDU/CLASSES/CS165/](http://daslab.seas.harvard.edu/classes/cs165/)

Prof: Stratos Idreos, SEC 4.411

*After taking CS165 you will be able to understand data system internals, code and design efficient data-driven algorithms and data structures, and have a handle on what it means to do CS systems research.*



# QUICK SUMMARY . CS165

*/\*State-of-the-art data structures, algorithms and data system design\*/*

*/\*Research based, Problem based, Semi-flipped classroom, OH & Labs daily\*/*

*/\*Additional sections provided online pre-recorded, Guest lectures from industry\*/*

*/\*Two take-home midterms with open books, laptops, and tablets (late Oct and Nov)\*/*

*/\*Large, open-ended, individual software project in C due at the end of the semester\*/*

*/\*Project Midway Check-in: small deliverable to warm up - hard deadline in mid October\*/*

*/\*Ideal background: basic data structures, algorithms, and systems programming (C/C++)\*/*

*/\*Expected workload: a lot - do not combine with other heavy courses\*/*

*/\*Grading: Project 60%, Class Part. 10%, Midterms 30% + bonus opportunities\*/*

*/\*We count best of Midterm2, and Midterm1+2\*/*

*/\*All info is on the class website (not Canvas), Class forum contains all announcements/updates\*/*

*/\*CS165 helps with software engineering job placement/interviews\*/*

*/\*CS165 helps with research initiation in algorithms/storage/systems/computational data science\*/*

*/\***For our extension school students:** unfortunately due to COVID restrictions, and unlike previous years, we cannot welcome on campus at all this semester extension school students even if they are local students; all meetings, office hours, sections and exams will take place online.\*/*

***/\*CS165 students have won the ACM SIGMOD undergraduate research competition 6 times in 2016, 2017, 2018, 2019, 2020, & 2022 \*/***

**Do:** Read Syllabus; Check self-evaluation guide, semester project; Project 0

**If you take the course:** Register on class forum, & notes; Get starter code / tests

**Success strategy:** Participate actively in all classes; Go to Labs/OH weekly



## WHAT IS THIS CLASS ABOUT?

We are in the big data era. Every two days we create as much data as much we created from the dawn of humanity up to 2003 [Eric Schmidt, Google]. We are going through major transformations in businesses, sciences, as well as everyday life - collecting and analyzing data, applying machine learning across diverse scenarios, and constructing large and complex data science pipelines change everything; the key common ingredient in all those cases is storing and accessing large amounts of data.



In fact, this is typically the bottleneck in any complex data-driven process/program because the way hardware has evolved, moving data is way more expensive than doing any kind of computation. Data systems provide the means to store and analyze a massive amount of data and thus they sit in the critical path of everything we do. A data system is effectively a large collection of data structures and algorithms that work in synchrony to solve complex data access requests. It is a \$100B industry, growing 10% every year [Economist, "Data, data everywhere"].

This course is a comprehensive introduction to modern data systems. The primary focus is on trends that are shaping the data management industry right now: column-store and hybrid storage systems, NoSQL key-value stores, shared nothing architectures, cache conscious algorithms, hardware/software co-design, main-memory systems, adaptive indexing, stream processing, and scientific data management. We also study the history of data systems, traditional and seminal concepts and ideas such as the relational model, row-store database systems, optimization, indexing, concurrency control, recovery and SQL. In this way, we discuss both how and why data systems evolved over the years, as well as how these concepts apply today and how data systems might evolve in the future. We focus on understanding concepts and trends rather than specific techniques that will soon be outdated - as such **the class relies on recent research material and on a semi-flipped class model** with a lot of hands-on interaction in each class.



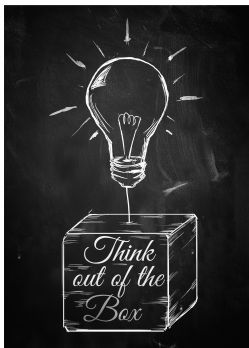
## WHAT IS THIS CLASS NOT ABOUT?

This class is not a traditional introduction on how we use a database system and how to write SQL. Instead, this is a data structures and algorithms class about data system design. You will learn how data systems work at their core and how to design new systems for emerging applications and hardware. By the way, if you know how systems work, you also become better at using them!



## WHY TAKE THIS CLASS?

Data systems research and the whole industry are going through a major and continuous transition; given that new data-driven scenarios and applications continuously pop up, **there is a continuous need to redefine what is a good data system design**. Every single big data-driven company, startup, government or scientific organization is limited by their data system choices and continuously seeking new solutions tailored to their problems. Given that there is no perfect data system, there is a constant need for deep expertise on designing performant workload and hardware conscious data structures, algorithms and complete systems.



CS165 exposes students to the core internals of data systems making it possible to understand core trends in data structure, algorithm, and system design and to be one of the very few who know how to design and evaluate systems. In addition, due to the way the course is taught with a focus on interactive problem solving, open topics and the latest research results, this is also a great class for those who want to understand what CS research is all about and how to engage in doing research. So far 11 undergraduate teams have made it to the finals of the ACM SIGMOD Undergraduate Research Competition with research projects that came out of the course. We have won first place in 6 years. For example, in 2016, we won first place with the work on Adaptive Denormalization; in 2017 we won first place with the work on Evolving Trees; in 2018 we won first place with

the work on Splaying LSM-Trees; and in 2019 we won first place with a project that shows it is possible to efficiently transition between LSM-tree and B-Trees.

## WHAT IS THE EXPECTED LEARNING OUTCOME?

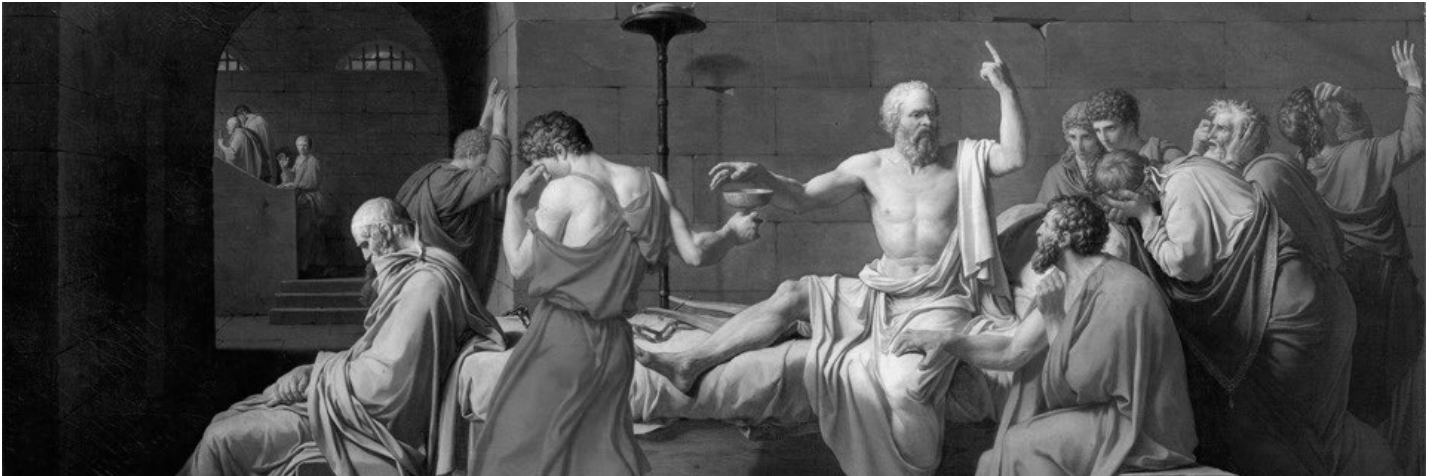
1. To become familiar with the history and evolution of data systems design over the past five/six decades (from relational to NoSQL).
2. To understand the basic tradeoffs in designing and implementing modern data systems and access methods through a step-by-step hands-on experience.
3. To be able to design a new data system given a data-driven scenario and build a functional hardware conscious prototype that is close to the optimal design.
4. To be able to understand which data system is a good fit given the needs of an application.
5. To deepen C programming, debugging, and performance profiling skills.



data system kernel



cs165/265 student



## CLASS PHILOSOPHY

CS165 has unlimited office hours, relies on the latest research papers instead of a standard text book, lectures are based on interaction and discussion instead of just “lecturing”, many of the quizzes and problem sets are actually open research problems and most of all it is fun! The instructor and the TFs are here to help you every day and at all times throughout the semester. You may request as many meetings as you like and as much help as you want.

The course is also geared towards engaging creative thinking and problem solving to give students a feeling of how computer science research takes place. Many of our students in the past have successfully engaged in research projects with [DASlab](#), published research papers, and won major international research competitions.

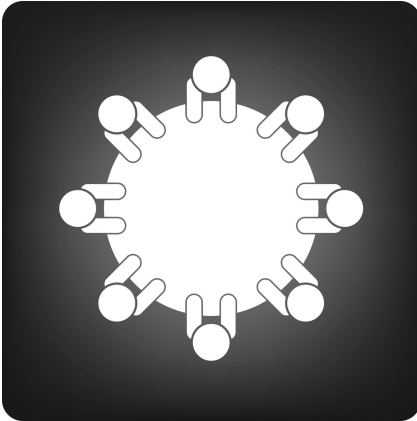
From your side, you should be aware that this is a demanding class that combines knowledge about system design, algorithm design, data structures and includes a non-trivial systems project. You are going to learn state-of-the-art techniques that are being applied in the real world right now. Following the material of the class and performing a successful project requires serious weekly commitment throughout the semester.





## LECTURES

The class meets twice a week: (check website for room and timeslots). Classes are designed to be discussion-based and slides are used to drive discussions as opposed to delivering the material.



## INTERACTION IN EVERY CLASS

In every class there will be a 20-30 minute session where students will work on problems in groups of 3-4 students. The instructor and the TFs will be walking around in the classroom to participate in the discussions and brainstorm with the students. The problems will be based on material that has been presented in class and these discussions will be used to either solve open problems or to introduce new ideas. The topics in our midterms will resemble the topics and expectations during those interactive sessions and we will also use those sessions to brainstorm about the milestones of the semester project.

## DAILY OFFICE HOURS & LABS

Interaction does not stop in lecture time. CS165 is designed to maximize interaction as we truly believe this is the best way to learn; we offer daily office hours and labs. Starting Week 1, Prof. Idreos will hold office hours every week day in his office. Labs are also offered every day of the week as of Week 2 by the TFs (check the class website to get the exact time slots and rooms).

*The goal of OH* is to provide any kind of feedback on the class material, e.g., about the design of data structures, algorithms and systems. You should also come to OH to discuss the design of your project and to get feedback on your design documents. You are also welcome to come to OH for any other general question regarding classes, careers in industry/academia, PhDs, etc.

*Labs:* The main goal of Labs is to provide hands-on help for the project. So bring your laptop and your questions about specific project parts you need help with. Labs are the place to go when you have a persistent bug, when you need help with a specific tool for the project (e.g., for debugging or performance testing) or to get feedback about the quality of your coding.

Finding and fixing bugs can be very difficult and time consuming. As such, we want to make the time you spend in Labs is as useful as possible. We want to teach you the process of finding and fixing bugs, not just solve a bug for you. We expect that before coming to labs you have spend several hours “fighting” a bug. Then if you cannot make any more progress on your own, you should come by and by then you will have enough experience to really understand the solution and the process. Do not feel like something wrong is happening if you find yourself stack with a bug for a day or two. This is normal and part of the learning process. It will and should happen several times through the semester. Before coming to discuss a bug you should perform/answer several questions on your own: Check the class website for exact instructions.



## ATTENDANCE & SIMULTANEOUS ENROLLMENT

The best way to learn is through discussion and interaction with the instructor and the TFs. Our classes are not about “lecturing” – they are semi-flipped and all about active interaction. We hope to see you there! There is a 10% “active class participation” grade component to encourage active participation in class discussions. **All classes will be recorded and will be available online.** Due to the active class components, simultaneous enrollment is not allowed in CS165. However, we may still do an exception for senior students depending on their exact situation if this is the last opportunity to take CS165.

## NO LAPTOP/PHONE POLICY

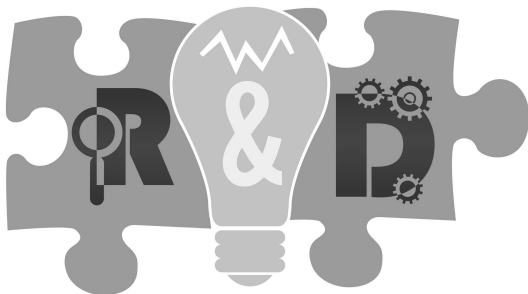
CS165 is based on interaction. We want students actively participating in class and interactive sessions, asking and answering questions to maximize learning. In each class, we will bring a printed copy of the slides for each one of the students so you can more easily follow along and to minimize note keeping. Recent studies show that even if you only use a laptop for note taking, it can have a negative impact on how well you understand the material in class<sup>1</sup>.



*[The Pen Is Mightier Than the Keyboard: Advantages of Longhand Over Laptop Note Taking. Pam A. Mueller and Daniel M. Oppenheimer. Psychological Science. 2014, Vol. 25(6) 1159–1168]*

## SECTIONS

Another component of the course is sections. Sections are used to deliver material about the class, i.e., to go more deeply into some of the concepts discussed in class, to do additional quizzes, or to deliver background material that is needed to follow next week’s class or for the project. There will be no actual section meeting. Instead, all sections will be recored by the TFs and videos will be posted online. The material posted will be tailored to present a step by step guide for any of the topics presented to make it easy to follow everything without having to be physically present in an actual section. However, if there are still questions about the material presented in sections, you will be able to ask those questions either during the daily office hours or during the daily labs.



## RESEARCH AND DISCUSSION MEETINGS

On select evenings the instructor, and DASlab PhDs and postdocs will hold meetings to discuss research! They will present their recent work on and connect it with the class material. Then, you will get the chance to talk with them about their research, open problems and be exposed to open research opportunities. In addition, we hold meetings to discuss careers in industry and academia, grad school and anything else you may have in mind.

---

<sup>1</sup> There are cases where having a phone or laptop during class is necessary such as when you expect an important call or message or when you need the laptop to better follow the slides due to any issues with your eyes or ears. Just let the instructor know and all such cases will be granted permission to use any tools necessary.



## REQUIRED TEXTBOOK

The class is about state-of-the-art data system design. There is no textbook for that. Thus, we use recent research papers and surveys which will be posted on the course website, which you will have access to through the Harvard network. We also use the following textbook: Database Management Systems, by Raghu Ramakrishnan and Johannes Gehrke. This textbook is a great source for all the seminal and traditional topics that we will cover.

## SLIDES/NOTES

The slides used during the course will be available online before each class. We will also print slides for you and bring them to each class. If there is material that we want to communicate to you only after class, this will be available shortly after each class.

**SLIDES ARE NOT NOTES!** You should not expect the slides to cover the material in detail. The class is based on discussion and problem solving; the slides are tailored to drive the discussion as opposed to serving the material.



In each class one or more students will be assigned to take notes. After class these students will populate a collaborative notes document and then all students are welcome to jump in and enrich the notes further. Collaborative note taking and editing will be part of your class participation grade and a great way to recite the material and also see how your fellow students perceive it.

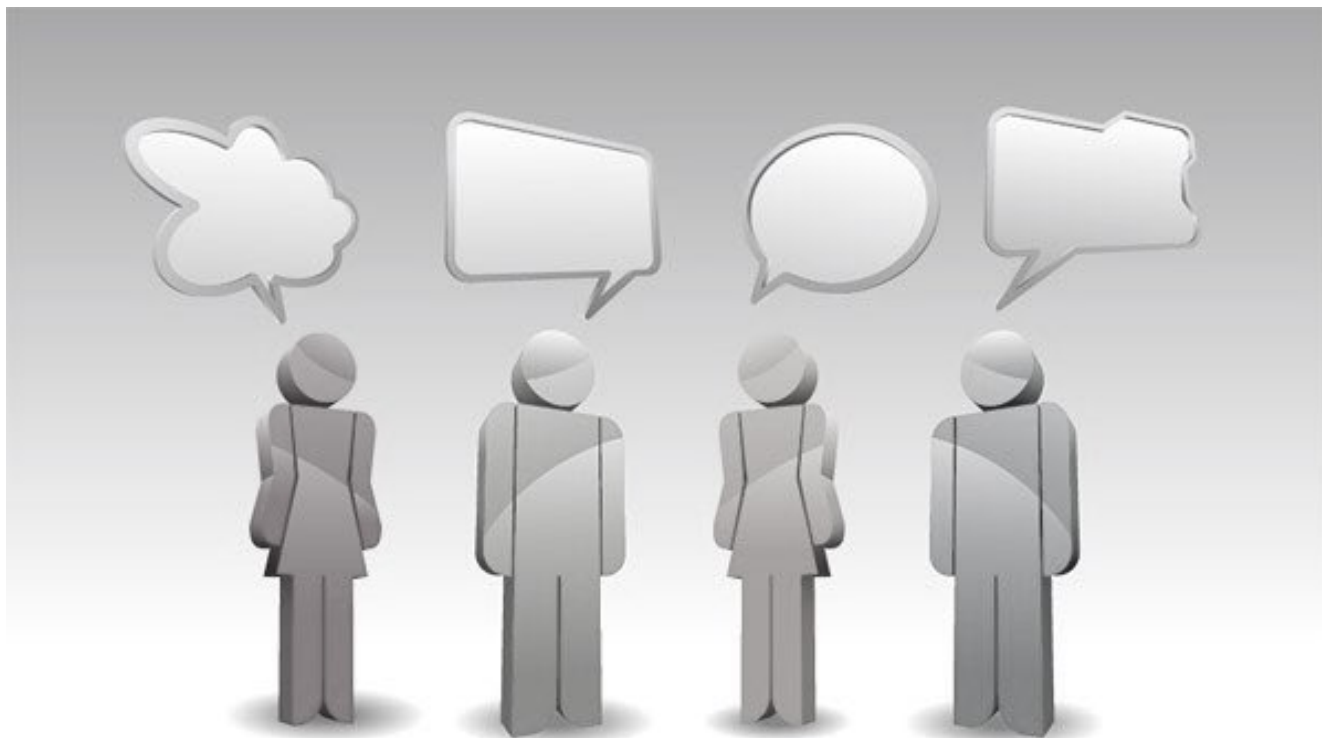
The link to the collaborative notes is available on the class website.

## GUEST LECTURES

Every semester we arrange a few guest lectures by leaders in data system design from industry and academia. Past guest lecturers include: Guy Lohman from IBM Research, Erietta Liarou from EPFL Lausanne, Alkis Simitsis and Georgia Koutrika from HP Labs, Nikita Shamgunov from MemSQL, Laura Haas from IBM Research, Nga Tran from Vertica, Jignesh Patel from University of Wisconsin, Magda Balazinska from University of Washington, Johannes Gherke from Microsoft, Goetz Graefe from Google, Marcin Zukowski from Snowflake, Justin Levandoski from Microsoft Research.

You will get the opportunity to both hear a guest lecture and to actively participate in discussions with our guest speakers.





## ONLINE DISCUSSIONS

We will use an online forum. The links are posted on the class website. We continuously monitor the forum and will be answering your questions promptly. At the latest we will give you an answer within a day but typically you can expect answers within a few hours. You are welcome to post any question that might help you understand the material better or help you with the project. Anonymous posting (to the other students) will be enabled so that students feel more comfortable posting questions. You should also use the forum to privately reach out the staff as opposed to using email.

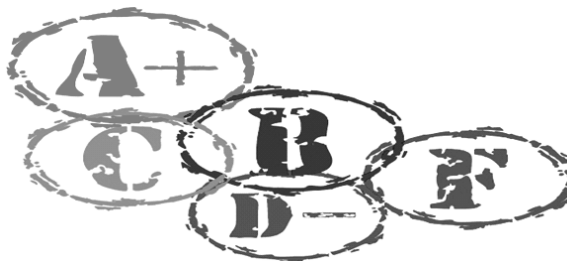
**BASIC RULES:** We only have a few basic rules so we can keep the forum functional and useful for the students as well as manageable for the staff.

- 1) We ask that you first **search the forum well before** posting a question so that we do not have duplicate entries and to make it easier for the teaching staff to focus on unresolved issues.
- 2) Please make sure to **stay on top of all staff posts** (especially those that are pinned). Anything we post on the forum, we consider as “known”.
- 3) **Do not use the forum to post code** and ask help with debugging. While it can work in some cases, remote debugging is a pain and takes a lot of time. We have labs every day. Bring your laptop and we will help you on site or join remotely and we will help you via a shared screen mode.
- 4) Before posting **consider whether your question should be public or private**. If this is about your own status or unique problem, post privately to minimize noise for the rest of the class. Typically all issues regarding technical clarifications or logistics can be public. **Any personal complains or concerns should be dealt with strictly during OH.**



## GRADING

- Active class participation & quizzes: 10%
- Midterm 1: best of 0 or 15%
- Midterm 2: best of 15 or 30%
- Project Tests: 20%
- Project Code review: 20%
- Project Final design/paper: 10%
- Project Midway Check-in: 10%
- Bonus: Extra project tasks: up to 5%
- Bonus: Speed prize: up to 5%



This adds up to more than 100%, however the grades are judged upon a 100% scale.

## PASS FAIL/ AUDITING

We do not allow pass fail in CS165. Due to the interactive nature of the course, for every student that takes it, the teaching staff need to invest a lot of time during class, OH and labs. We expect students to fully commit and we are here to help you all the way through every single day. We may allow a couple of audit slots depending on the number of students.

## MIDTERMS

We hold two midterms. Midterms start in class so we can answer any possible question and then students can take the exam home to complete within 24 hours. Students are allowed to use any notes, books, and class materials. Of course, we expect that everyone completes their midterm independently.

Midterms are not designed to test how much you can remember from the content. Instead, they stress your ability to come up with new solutions, think through all design decisions and side effects of any solution you choose, and how you communicate your design.

The best way to prepare for midterms is to have an excellent handle on all the topics we work on during our interactive in-class sessions.

not have to study for midterms alone. Before each midterm, we hold special office hours and sections to help with preparation. After every midterm, we hold special sections to go through the solutions.

Due to the open problem-solving nature of the Midterms that for some students might require some adjustment, we give the option of not grading Midterm 1. The final Midterm grade will be the best of:  $15\% \text{Midterm1} + 15\% \text{Midterm2}$  and  $0\% \text{Midterm1} + 30\% \text{Midterm2}$ .





## CS165 WICKED AWESOME SEMESTER PROJECT

Design and build a main-memory optimized column-store

**Project Website:** <http://daslab.seas.harvard.edu/classes/cs165/project.html>

The class has a running project throughout the semester.. The project is about designing and implementing a prototype of a modern main-memory optimized column-store data system. By the end of the project you will have designed, implemented, and evaluated several key elements of a modern data system and you will have experienced several design tradeoffs in the same way they are experienced in industry labs.

**This is a challenging but fun project!** We will also point to several open research problems throughout the semester that may be studied on top of the class project and that you may decide to take on as a research project. The project has a total of five milestones with specific expected deliverables.

**The five deliverables are:** 1) basic storage layer, 2) indexing methods optimized for main-memory, 3) shared scans methods, 4) joins, and 5) updates.

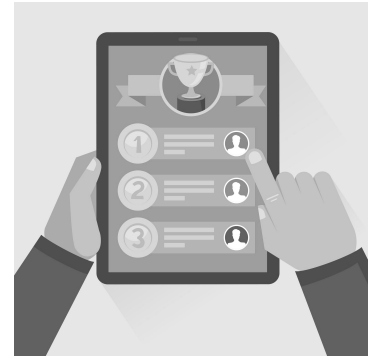


The deliverables will be tested using predefined automated unit tests for functionality and, as extra credit, for performance.

**Automated Testing Infrastructure:** We have an automated testing infrastructure. We provide a series of tests (using both fixed and randomized data) to automatically test your code for each project milestone. You are able to submit your code daily and get results by automated emails overnight. Tests run against an in-house Linux server at DASlab. You will be able to find the exact specifications of the machine and tests on the project website. Once you pass all the tests in the testing infrastructure your project is complete!



**Leaderboard:** We will have a running competition and an anonymous leaderboard so you can continuously compare your system's performance against the rest of the class (and past classes). Essentially we provide additional tests that increase the amount of test data so performance differences between projects will be highlighted. You will be able to run these tests daily as well, so you can improve throughout the semester. We will also provide a "benchmark" entry in the leaderboard which represents what we consider good performance for each milestone based on an in-house implementation from the lab.



**We will give you starting code** that implements the basic client-server functionality (i.e., communication) so you can focus on building the server side code, that is, the essential core data processing algorithms and data structures of a database system. In addition, whenever applicable we will let you know if there are existing libraries that is OK to use.



**Evaluation:** Individual deliverables should pass all provided tests on the testing infrastructure. However, you will not be judged only on how well your system works; it should be clear that you have designed and implemented the whole system, i.e., you should be able to perform changes on-the-fly and explain design details. At the end of the semester each student will have a 1-hour session with the instructor and another 1-hour session with the TFs where the student will demonstrate the system, and answer questions about the design and about supporting alternative functionality. [Tip: From past experience we found that frequent participation in office hours, brainstorming

sessions and labs implies that the instructor and the TFs are very well aware of your system and your progress which makes the final evaluation a mere formality in these cases.]

**Collaboration Policy:** The project is an individual project. The final deliverable should be personal. You must write from scratch all the code of your system and all documentation and reports. Discussing the design and implementation problems with other students is allowed and encouraged! We will do so in the class as well and during office hours, labs and brainstorming sessions. All students that have collaborated with other students in whatever capacity should provide a collaboration statement with their final deliverable to properly acknowledge any ideas that was taken or was influenced by discussions with other students.

**Late Days Policy & Schedule:** The project is due at the end of the semester. In the project description you can find a detailed time-schedule that we propose you follow. With the exception of the midway check-in (which is a hard deadline), the rest is a "suggested schedule" that will allow you to spread the work throughout the semester and to have sufficient time for each milestone based on the complexity and the work required at each phase of the project. This is an involved project that requires commitment through the entire semester and cannot be done in 2-3 weeks at the end. Not submitting the project milestones on time will have no side-effects on your grade but at the same time, we will not be able to provide you with any feedback on your progress until we have your design documents and your code.



**Note:** Experience says that every year a number of students cannot handle the freedom to self-pace, and end up significantly deviating from the schedule. We will send you frequent reminders but you should know that deviating from the schedule by more than a couple of weeks will most likely mean that you will not be able to finish the whole project by the end of the semester (unless you are already an experienced systems student).

**Midway Check-in:** The goal here is to demonstrate that you are having decent progress and mainly to avoid falling behind. By late October each student should 1) deliver a design document that describes the intended design for the first two milestones (5%) and 2) have implemented a project that passes at least the first three tests of the first milestone in the automated testing infrastructure (5%). A template of the expected design document is provided online. **The midway check-in deadline is a hard one;** no extensions will be given so please do not ask for one unless you think there is a fair reason such as a medical issue. The reason is that we are trying to make students see the scope of the project early on.

**Speed Prize:** The three fastest projects (top 3 in the leaderboard by the end of the testing period) will gain extra points (5%). The competition will terminate the last day before we need to upload grades so you will have plenty of time to improve (until around mid December).

**Extra Points for Bonus Tasks:** We will regularly assign extra tasks or you can come up with your own extra tasks for the various components of the project. With these extra tasks you gain extra points (up to 5%).

**What is a Successful Project?** A successful project passes all the predefined tests we provide on the testing infrastructure and the student successfully passes the final face-to-face evaluation. A successful final evaluation is one where the student is able: (1) to fully explain every detail of the design, and (2) to propose efficient designs for new functionality on the spot. On the class website you will find a step-by-step guide that will help you prepare for the evaluation meeting.



## FEEDBACK ON PROGRESS

We provide feedback continuously. The main thing that you will need feedback on is your semester project. The way to get feedback is to show up to our daily office hours and labs and share your design decisions, code, and test results with the staff. In this way, you will get hands-on help and feedback. Feedback on midterms will be provided within two weeks and you are welcome to come by during office hours to discuss any one of the tasks. We also welcome feedback and ideas about the course at any point during the semester. Just come and chat with us during office hours! Tell us how you are keeping up and how we can make it easier for you.





## WHO CAN TAKE THIS CLASS?

You probably heard stories that this is a very heavy class and that the project will consume a ton of your time. While this is true, it is also true that you will have a lot of help! So fear not.

**Background:** Naturally, the more background you have the smoother your experience in 165 will be. Prior knowledge of C programming and systems programming, as well as a good understanding of computer architecture and in particular the memory hierarchy (cache memories) is very important for this class. Courses providing systems background (like CS50 and in particular CS61 or equivalent) are essential. Good hacking, algorithm designing, and data structures skills are also required.



**A self-evaluation guide** is posted on the class website to help you understand if you qualify for the course and how much material you might need to cover. The course (lectures, sections, labs, and office hours) is designed so you can acquire the necessary background even if you are missing some essential knowledge at the beginning of the semester. So we have you covered. However, you should be aware that if you did not breeze through the self-evaluation guide you will have to put in more hours to successfully complete the course. Talk to the instructor if you have not taken CS61 or if you do not feel completely comfortable with the self-test but you still think you are ready for CS165.

**Project 0:** We provide a Project 0 that is designed to 1) help you get an idea about how fit you are for the class and 2) bootstrap your semester project. Essentially Project 0 consists of an independent data structure design and implementation in C that you can later on use (nearly) as is for the first milestone of your semester project.

If you are reading this text a few weeks or even months before the semester starts, you can use the guidelines on the class website to prepare for the course. There you will find specific study material and programming exercises.



## HOW CAN I DO GREAT IN 165?

Just utilize all resources provided. Show up in class to participate in interactive sessions. There are also daily office hours and labs; show up as often as possible so we can help with anything you need! When you find yourself stuck with the project either with a design decision or just a bug, it is normal to struggle for a while — it is part of the learning process — but after some time grab your laptop and come by!



## PLAGIARISM

If we suspect that a student has cheated in any way they will be reported immediately. We use code analysis tools to compare the code that each student submits against all other projects that have been submitted. This includes all projects of all past years and excludes the starter code. In addition, we perform in person code and design reviews at the end of the semester where it is easy to spot suspicious cases: we expect students to truly own every single line of code and every single design decision.

You are responsible for understanding Harvard and Harvard Extension School policies on academic integrity ([www.extension.harvard.edu/resources-policies/student-conduct/academic-integrity](http://www.extension.harvard.edu/resources-policies/student-conduct/academic-integrity)) and how to use sources responsibly. Not knowing the rules, misunderstanding the rules, running out of time, submitting "the wrong draft", or being overwhelmed with multiple demands are not acceptable excuses. There are no excuses for failure to uphold academic integrity. To support your learning about academic citation rules, please visit the Harvard Extension School Tips to Avoid Plagiarism ([www.extension.harvard.edu/resources-policies/resources/tips-avoid-plagiarism](http://www.extension.harvard.edu/resources-policies/resources/tips-avoid-plagiarism)), where you'll find links to the Harvard Guide to Using Sources and two, free, online 15-minute tutorials to test your knowledge of academic citation policy. The tutorials are anonymous open-learning tools.

**Should you find yourself in trouble, falling behind in any way and for whatever reason, please do not cheat - instead, talk to the instructor and we will help you in any way we can.**

**If you did cheat, e.g., because you were under stress, come forward within two days (e.g., of submitting the code or the midterm); we will resolve the issue internally and try to help.**

## ACCESSIBILITY

Harvard and the Extension School are committed to providing an accessible academic community. The Disability Services Office offers a variety of accommodations and services to students with documented disabilities. Please visit [www.extension.harvard.edu/resources-policies/resources/disability-services-accessibility](http://www.extension.harvard.edu/resources-policies/resources/disability-services-accessibility) for more information and please contact prof. Idreos directly with any questions or concerns you might have. We will help in any possible way to make your experience in this course a great one!



## EXTENSION SCHOOL

Lectures will be broadcasted live. Lectures will also be available for on-demand broadcast within 24 hours after each class. Students will be able to watch the live or recorded broadcast through their browser using the Matterhorn player. The link to the broadcasts for CS165 will be available through the Canvas website for this class and will also be posted on the class website before the first lecture.

**Capturing Discussions:** Given that a big portion of the class is based on interaction, extension school in cooperation with the class staff is working to set-up a system with several microphones across the classroom so we can accurately and clearly capture brainstorming discussions and comments during class time. Microphones will “follow” the instructor.

**Class Participation:** Extension school students will be able to participate live in classes using Zoom. In every class there will be one TF assigned on Zoom and extension school students will be able to ask questions directly via chat. In addition, during quiz time there will be an extension school group headed by a TF to work on the quiz live where microphones will be enabled.

**Office Hours and Labs:** For remote students we will have special slots, including weekend slots, that will be accessible only remotely with Zoom. Other than standard chatting and talking features, Zoom offers screen sharing which can be used when you need help with specific issues such as debugging. **While OH and Labs are optional, they are an essential part of the course, and you should plan to participate nearly every week to avoid falling behind.**

**Midterms:** Midterms are take-home exams that are submitted through Canvas.

**Online Forum:** To participate in piazza you need a Harvard email address. If you do not have one you can create one here: <http://www.extension.harvard.edu/resources-policies/resources/computer-e-mail-services>. Note that there is a separate forum for extension school students.

**Grading:** Even though we encourage extension school students to utilize the opportunity to interact with the staff and participate in class live we know that for practical reasons this will not be possible for all students. For this reason there will be no “class participation” grade and the portion of this grade will be distributed equally in project (65% including Midway check in) and midterms (35%). Students that do participate actively can choose to be graded with class participation.

**Course Rigor:** Other than class participation, the requirements and expectations are exactly the same across college and extension school students.

**Graduate Credit:** Extension school students who take the course for graduate credit should provide a detailed literature review of NoSQL key-value stores. This is due at the end of the semester along with the project and will account for 20% of the project grade.