

Course Syllabus, Fall 2023

Welcome to **CSCI0200**, Brown's second course in the CS intro sequence. This syllabus outlines the course structure, requirements, and policies. It will be the definitive source for policy questions.

High-Level Course Information

Time and Location: MWF 2 - 2:50pm, Sydney Frank Hall Room 220 (Marcuvitz Auditorium) and recorded to Panopto

Professor: Nick DeMarinis (cs200-profs@brown.edu)

Prerequisites: CS111+bridge approval, CS112, CS15, CS17, CS19 or permission of instructor

Description: Students extend their program-design skills while learning multiple data structures, common graph algorithms, different forms of societal impacts from programs, how to analyze programs for performance, and how to work effectively with multiple styles of programming languages. Examples and course projects draw from several areas of computer science to help students identify their broader interests within the field.

Time Requirements: In addition to three hours per week in lecture, you will spend 2 hours a week in lab and 10-15 hours on homeworks and projects (the higher end is for projects; homeworks should take less time). Time expectations are the same for those in the remote/asynchronous section: you simply watch the recorded lectures (and try the in-lecture activities) on your own schedule, while still attending a lab section (perhaps virtually, but still synchronously). The schedule of assignments and deadlines is on the assignments page of the course website.

Course Website: The lecture schedule, notes, deadlines and all assignments will be online at <https://brown-csci0200.github.io>. We will only use Canvas to distribute files and videos restricted to members of the course.

Head Teaching Assistants:

Elsa Choi-Hausman, Paul Lestz, Yasmine Abdelaziz

Teaching Assistants:

See the [staff webpage](#)

The Course At A Glance

More details on these points appear later in this document.

- Lectures MWF at 2pm in Sydney Frank Hall 220 (Marcuvitz Auditorium, aka MARC)
 - There will be lots of in-class exercises to make real-time participation worthwhile
 - Lectures will be **recorded** (files in Canvas Media Library), but not designed for livestreaming
 - We'll be taking real-time Q&A on Ed during lectures
 - If you want to take another course at the same time, request an override for section S02 (which has no fixed time)
 - For the **lectures 2-6 (Sep 8 - Sep 18), we'll be split into two tracks** (one for those coming from 15 with Prof. Kathi Fisler, one from those from 111/112/17/19 with Nick). We'll release scheduling details after the first lecture.
- Everyone attends a weekly 2-hour lab section with roughly 1 TA per 10-12 students.
 - You'll **sign up for lab times during the first week of courses** (adjusting after shopping period if necessary)
 - You can miss two labs during the semester without penalty
- An assignment (either a one-week homework or stages of a 2-week pair project) will be due every week (on Friday evenings; this schedule will be slightly different for the first two weeks). Only one of these will be out at a time.
 - You have a budget of **10 late days** to use on assignments and projects (max 3 per assignment) across the semester. These cover travel, deadline pressure, and minor illness. **Once you have used your late days, we will not grade your subsequent work.**
 - Approved extensions (see next major bullet) will not count against late days.
 - Work may not be submitted more than 3 days after the corresponding assignment deadline without an approved extension.
 - Regrade requests will close for each assignment one week after the grades are posted for that assignment.
 - No work will be out over Thanksgiving Break
- **Extension requests** must be submitted via the form on the course homepage. Only professors may grant extensions.
 - **Extensions require documentation** such as a dean's note, health services note, loaner laptop checkout, or something similar. Heavy workloads in a given week are not grounds for an extension.
 - Religious observance that goes beyond 24 hours is valid grounds for an extension. No additional documentation is required in this case, though the professors may confirm dates and practices with the Brown Chaplains' Office. If you feel you have an extenuating circumstance

- that does not fit the above justified reasons, reach out to Nick (not the (H)TAs).
- **TA Office hours:** will consist of group-style sessions where TAs will circulate around a room providing conceptual and debugging help to many students in a collaborative setting.
 - How it works: all office hours will be held in a lab or conference room—just show up and start working. TAs will move around and take questions on debugging or concepts to help point you in the right direction. When there are conceptual or other questions that apply to many students, TAs will switch to “teaching mode” and talk so everyone can listen in. We are trying this new format to maximize how much time we can spend working with you, and we encourage you to work with other students to maximize your own learning.
 - Our collaboration policy is designed to provide opportunities to work with others on conceptual or debugging problems (details below)
 - **We will not hold open individual debugging hours—most debugging problems should be resolved in group hours.** If a particularly difficult problem cannot be resolved this way, we will talk with you to discuss next steps (posting on Ed for the whole staff to see, setting up appointments). Otherwise, if you are stuck to an extent that group hours aren’t helping, contact Nick for an appointment.
- Nearly all **communication should go through Ed**. Except:
 - for sensitive or personal matters that HTAs should not see, email Nick at cs200-profs@brown.edu. This includes anything regarding SAS accommodations
 - **For fastest processing, please don’t use Nick’s personal email addresses** (the separate inbox helps Nick stay on top of course-specific mail)
- There will be a **written midterm** (Tues Oct 24, 7-9pm) and a **written final** on Saturday, December 16 at 2pm. **Barring a documented emergency or being off-campus for the semester, you must take the exams in person on one of the available dates.** Having already bought plane tickets home is not a justified excuse from the in-person final; in these cases, you can get an incomplete and take the final with the next offering in Spring 2024.
- The [collaboration policy](#) allows discussion with peers on concepts and low-level bugs, but you must develop your code on your own (unless an assignment states otherwise).
 - You can help each other with IntelliJ setup issues, syntax, and understanding where bugs are coming from (but you’ll fix your code on your own)
 - You’ll submit a collaboration statement with each assignment telling us who you talked to for conceptual and debugging help.
 - If the staff are suspicious regarding whether you did your own work, we reserve the right to ask you to come in and explain your work.
 - You have 24 hours from the time you submit an assignment to admit to violations (receiving lower penalties unless the violation is egregious); violations that we uncover after 24 hours are subject to penalties ranging from a loss of letter grade in the course to a directed NC, depending on the circumstances.

Diversity and Professionalism

Every student has the right to a supportive learning environment in this course. In a supportive environment, all students are treated with respect and considered able to learn, regardless of their culture, gender, beliefs, abilities, or prior CS experience. Unsupportive environments distract students from learning, often by trying to make people feel uncomfortable about who they are or whether they belong. We all share the responsibility for creating a supportive environment in this course. While some of this responsibility lies with the course staff, much of it ultimately rests with you, the students. *Nobody has the right to diminish the educational opportunity of another person who is participating in this course.*

So what are we asking of everyone?

Don't assume that someone with a question is stupid or representative of anyone but themselves:

There are any number of reasons why someone might need more time to grasp an assignment or a concept. Maybe they first started programming last semester. Maybe they just bombed a test in another class. Maybe they are distracted by family concerns (health, finances, etc). Maybe they've had a fight with a partner. Maybe their brain just locked up. Be kind. Everyone got into Brown. Everyone deserves to be here (both at Brown and in CS0200). Look for ways to help each other learn, not knock each other down.

Don't make assumptions about your classmates' backgrounds, situations or beliefs: Not everyone has a fast laptop, a love of playing video games, prior programming experience, or high grades because of the culture they appear to be from. Not everyone at Brown holds progressive or liberal political opinions. An assignment that seems "so easy" to you might not have been for someone else. Try not to make generalized comments in public course spaces (like labs or Ed) that assume something about what your classmates have or have experienced. A student you aren't interacting with might overhear it and become self-conscious to the point that it adversely impacts their own work (this is a known phenomenon, called stereotype threat).

Respect that we're living in tough times for many: COVID, remote learning, international affairs, and the ongoing tensions in the USA about race and politics are impacting many of us (including on the staff). Some of us are dealing with significant disruption, anger, angst, fear, or threats to self or livelihood. Again, be kind and focus on helping each other learn, no matter what we're each going through.

Oh, come on ... This said, there are students in every graduating class for whom "being kind" isn't a compelling motivation. In the thrill of new-found liberation in college, or the desire to exert your emerging adult selves, some of you will cross the line, whether through "jokes" that aren't funny or other intimidating behaviors that expect others to "show a thick skin". To you, we say:

Be an adult and a professional: In the adult world, many of these line-crossing behaviors have a legal name: harassment. Let there be absolutely no doubt about this: harassment is against the law and it is completely against the norms by which we want to run this course and this department. (See [Brown's Title IX Web site](#).) We — the university, the department, and this course's staff — have absolutely zero tolerance for it. Workplaces are increasingly taking stronger actions against such behavior. In light of recent reports about such issues on campus, Brown is taking additional steps to reduce this form of harm. Therefore, if we cannot appeal to your decency, intelligence, and collegiality, let us at least appeal to your self-interest. Do not mess around on this matter. It will not go well for you.

However, we prefer that you think of this in positive terms. Your classmates are your colleagues. Someday you may be each others' start-up partners or co-employees; one of you may even be the other's interviewer or boss. So start treating one another like professionals, and we mean that in the best possible interpretation of that phrase.

In short: Be safe, be happy, and have fun without taking away anyone else's.

If you are experiencing difficulties or learning disruptions, we strongly encourage you to get in touch with someone so we can try to address the issue.

If we slip up, let us know: Professionalism and respect for diversity are not just matters between students; they also apply to how the course staff treat the students. The staff of this course will treat you in a way that respects our differences. However, despite our best efforts, we might slip up, hopefully inadvertently. When we do, please feel free to talk to us (profs or a trusted TA) about it. We promise to listen and to take your concerns seriously.

Sometimes, you may not be comfortable bringing issues up directly to us. If so, you are welcome to talk to Professor Tom Doeppner (director of undergraduate studies in the Computer Science department) or to Roberto Tamassia (the department chair). You of course may also talk to the deans, student support services, the staff in the Title IX office (which has a list of campus [resources](#)), or any other trusted adult on campus. Please don't suffer in silence though. Reach out to someone.

Textbook, Software, and Materials

There are no required textbooks for this course. You also don't need to purchase any software; the programs you write will all be written using publicly available and free software. Most assignments will require you to work on a laptop or computer, whether your own or ones in CS Labs (SunLab, etc). A Chromebook will not be powerful enough for some assignments.

We will be working in Java in the first part of the course and Python in the second. The official course tools are IntelliJ for Java and VSCode for Python. [This IntelliJ setup guide](#) or [this VSCode setup guide](#) will help you get these installed.

If you already have different preferred tools for working in Java or Python you are welcome to use them. You will need to be able to figure out how to work with JUnit 4 (for testing Java) and pytest (for testing Python) in whatever alternate environment you might be using. The staff will only help with installation issues in IntelliJ and VSCode.

We use Python 3 and Java version 19 or later. If you are **coming from CS15**, you'll need to upgrade your Java version so that you have the features we'll use for this course. If you are **coming from 111**, you will have to go through the VSCode setup again, as we use different Python packages in 200 than in 111.

Assignments Overview

Homeworks

Homeworks consist of programming tasks and written questions. Unless a handout explicitly says otherwise, you must write your code on your own on all homeworks, but you may discuss conceptual ideas with others (as long as you don't look at code while you do so). If you are mostly done with your code but are stuck on a bug, you may ask a classmate who is also mostly done to see if they spot the error. The intent is to prevent either of you from gaining insights in how to write the code from the other, while respecting that sometimes you just need another set of eyes to find a simple typo or thinko.

Projects

Projects are two-week (more open-ended) homeworks which you do in pairs; they include a required intermediate design review with a TA (which both partners must attend). You may find your own partners or have us match you with someone. You may work with the same partner on only one project. There will be two projects, starting roughly one month into the course.

Should a project team be unable to work together effectively, or should one partner end up having to carry the project weight for the other, the staff will split the pair. Depending on the circumstances, each split partner will either be asked to finish a portion of the project on their own or be allowed to join with another group. If you find yourself in a non-functioning partner situation, reach out to the HTAs and we will work out a solution around your circumstances.

As a general rule, you should not expect to receive credit for a project that you did not participate in completing. We encourage partners to keep track of who is doing what (perhaps in

email or text threads) to help us sort out claims that one person has done all of the project work.

Labs

Labs are weekly two-hour sessions in which you practice and check your understanding of course content (usually in preparation for homeworks and projects). Lab work is done with a partner. There will be roughly 20 students and two TAs per lab session. Each lab features checkpoints at which you'll need a TA to check your work so far. Once you complete the final checkpoint at the end of the lab, you are free to go. Labs are graded on active participation (actually working on the lab problems) rather than completion. It is okay if you don't finish everything during the lab time, but if you are routinely struggling to finish the labs, come talk to us so we can figure out why.

You do not need to complete unfinished lab work outside of lab time.

There is no way to do labs on your own time and get them checked off (we have allocated our staff resources elsewhere). You may wish to look over and refer to handouts for labs that you miss, as sometimes we use labs to introduce side concepts that are useful for projects. The "two missed labs" is meant to cover situations when you can't make lab. You will lose one percentage point off your course grade for each lab missed beyond the allowed two.

Students with prolonged extenuating circumstances should contact the professors regarding possible excused absences beyond the allowed two.

Exams

There will be two written exams: a midterm and a final. Both exams will test conceptual understanding, not programming. Exams from earlier offerings will be provided as the dates get closer. Exams will need to be done entirely on your own, with no help from others.

Exams will be held at the following times:

- **Midterm:** Tuesday, October 24 7-9pm EST (room TBA)
- **Final:** Saturday, December 16 at 2pm EST

Barring extreme circumstances (such as a health or personal emergency), **you must take these exams in person**. Do not purchase plane tickets for before the final exam. If you fly home early, you may take the final exam with the offering in spring 2024.

Students who are remote the entire semester will make separate arrangements with the professors. Students with SAS accommodations will get whatever is recommended in their official letter.

Grading

In CSCI0200, everyone who earns an A gets one. Your grade is independent of the grade or work of any other student in the course.

Grading in CS200 is designed to help you make progress in four core skill areas: Understanding and Choosing Data Structures and Algorithms, Programming and Program Design, Testing and Validation, and Technical and Social-Impact Analysis. Within each of these themes, we have identified [concrete learning objectives](#) that you are working towards (and that the assignments will be measuring). The rubrics for grading homeworks will frame your feedback around these four themes. We hope this will help you focus on your skills development in the course.

Initial course grades will be computed based on the following weights:

- 50% homeworks (later ones weighted more than early ones)
- 12% partner projects
- 8% labs (based on working actively through the lab period)
- 30% exams (combined)

Homework grades will be based on performance across the theme areas. Having some skills overall come out at the lower grade is more likely to result in your getting the lower grade. (For example, you can't ignore the testing component and rely on your programming portion to offset testing.)

We will also look for cases where later grades are stronger than earlier grades. This is designed to help students whose understanding improved as the course progressed.

In addition, a passing "average" across the midterm and final (combined, not separately) will be expected in order to pass the course. Basically, this means that you can't rely on homeworks and projects alone to pass the course. We will advise on what "passing" means after we see the results of each exam (but it typically comes out around the upper 50s).

We do not publish grade boundary cutoffs because grades are the boundaries depend on the overall pattern of your grades in the course. We will give guidance on how grades are progressing once we get into the course and have some data on which to base information.

Grading Your Code

We will grade your code partly by running it through an autograder. This will run your code against a test suite that we have developed. For this to work, your code needs to use the same method names and input/output types that are listed in the handouts. The stencil/starter code

for each programming-based project will contain a file named `AutograderCompatibility`. Running your code with this file included will check whether your code is consistent with these requirements.

Grading Your Test Suites

After the first couple of assignments, we will grade your test suites by running them through a different autograder. In this autograder, we will use your tests to check several “solutions” (some correct and some incorrect) that we have written. Your tests should pass on our correct solutions and fail on our incorrect ones (because your tests were thorough enough to catch the problem in our incorrect solution). You’ll see the terms *wheat* (correct solution) and *chaff* (incorrect solution) in some of the handouts. Your tests don’t have to detect every chaff, but they should detect most of them. More details will be on the individual assignments.

Regrade Requests

If you believe we made a mistake in marking your work against the rubric in Gradescope, file a regrade request on that specific question in Gradescope. We will accept regrade requests for one week after we publish grades on an assignment. It may take us a few weeks to process your request, depending on what other demands are falling on the staff in any given week.

Course Policies

Collaboration Policy

The CS200 collaboration policy is described in a [separate document](#). That document includes policies on using tools like CoPilot and ChatGPT for CS200 assignments.

Hours Policy

This year, CSCI0200 is conducting TA hours in a group format called “collaborative hours”. These are open hours where you can show up with conceptual or debugging questions. TAs will circulate around the room and provide help: depending on the number of students present and the questions, they will switch between talking about concepts, or providing debugging help.

You can use collaborative hours if you have specific questions, aren’t sure how to get started, want to talk about a specific problem in your code, or just want to work around others. Please just come! If you have a personal sensitive question you’d rather not discuss in a group format, please email Nick at cs200-profs@brown.edu to set up an appointment.

TAs have a broad discretion on the format for collaborative hours, in order to best support those who are present. For example, if there are a lot of common questions, TAs may talk about concepts at a whiteboard or projector. If everyone is debugging, TAs will move around the room looking at code and offering help on a first-come-first-serve basis. Most likely, both of these will happen at the same time (with multiple TAs).

When debugging in group hours, TAs will generally spend a few minutes with you to understand the problem and offer help to get you “unstuck” so that you can continue working/debugging on your own. The idea here is for TAs to be there when you need them, but to ultimately help you learn how to resolve problems. It also allows TAs to circulate around the room faster—that way, if you get “stuck” again, the TA can come back around!

If you need debugging help, we expect you to come to hours prepared to (a) show us the situation that you are stuck on and (b) explain what you’ve figured out so far about what is and isn’t working in your code. This will help us diagnose problems and offer help most efficiently. Ultimately, TAs will help you diagnose the problem and suggest how you can resolve it—you are responsible for correcting problems on your own.

Late Policy (including late days)

In general, our late policies are designed to give you some flexibility, while still allowing the staff to grade work in a timely manner. See the following sections for how this applies to each assignment, and about how to contact us regarding accommodations or extenuating circumstances.

Labs

Everyone is expected to attend lab every week (unless it is announced that there is no lab that week). Labs are where you get to practice material or learn concepts designed to help with upcoming projects; they are also where the TAs can get a sense of how you are doing in the course. Labs are designed to take the full two hours, but if you complete the final task and have been checked off by a TA before then, you are free to leave early. You will receive full credit for the lab if you are present and making progress on the lab throughout the lab time (even if you don’t finish all of the problems). Students who arrive more than 20 minutes late will not receive credit for that lab.

If you are unable to make your lab section during a particular week, fill out the lab switch form (on the webpage) **at least 24 hours in advance** to request an alternate time for that week. There is no guarantee that the HTAs will see a request filed closer to your regular lab time.

You can miss two labs during the semester without penalty (or need to make them up). You cannot make up a missed lab for credit (we don’t have the capacity to check you off, and checkoff is the foundation for getting credit for your lab work).

Homeworks and Projects

You have an overall **budget of 10 late days** that you can use to extend homework and project due dates (not labs or exams) across the semester. At most 3 late days can be used on any single assignment (so something due on Friday must be submitted by Monday), unless you have a separate extension granted by a professor.

You do not have to ask permission to use late days; we will automatically apply them at the end of the semester.

Once you have used all 10 late days, we will only grade subsequent work that is submitted by the original assignment due date.

On pair projects, late days are deducted for both partners.

Extensions

Your late-day budget is intended to cover most situations, including travel for interviews/athletics/conferences, heavy workload periods (e.g., other midterms), short-duration religious observance, and minor illness. If you have a multi-day religious observance or more serious situation as (will be) documented with a dean's note or health-services note, you may request an extension on an assignment (which would then not count against your late day budget).

Request an extension by filling out the extension-request form (linked on the course homepage). **Please use the form instead of emailing the instructor directly** (submitting the form will automatically generate a notification email to us). In a class this size, we are going to get a lot of email, and we don't want to lose track of your request or its granting. The form will keep all extension information in one place to prevent later confusion. *Only a professor (not the UTAs or HTAs) will be able to see the information you submit with your request, so as to protect your privacy.* The (H)TAs are not allowed to grant extensions.

Accommodations

If you feel you have physical, psychological, or abilities that could affect your performance in the course, we urge you to contact [SAS](#) (formerly SEAS). As a general rule, we make our best effort to support all accommodations recommended by SAS.

If you have accommodations beyond extended time or reduced-distraction spaces on exams, please email Nick to let him know what you might need. A few weeks into the course, they'll pull all the CSCI0200 accommodation letters from Brown's systems (which is why you don't need to tell us about typical time extensions).

If your accommodations indicate that you have a condition that might need you to get last-minute extensions on some assignments, you can request those by filling out the extension-request form when your situation arises. Your request will be granted, but the form will help us track that you had an extension (when it comes to grading, etc).

Unless there are special circumstances underlying your accommodations beyond these two cases, we don't need to set up a meeting about them. If you would prefer to meet or have an unusual situation, reach out to Nick to set up a time to talk.

Where to Get Help When You Need It

We are excited to help you continue learning CS in this course! Part of your learning experience in CSCI0200, and throughout your time at Brown, will involve some amount of productive struggle with problem solving. If/when you encounter feelings of frustration, don't be afraid to ask for help. Rather than beat your head against a wall, seek out your friendly staff. Sometimes, we can identify strategies you are or could be using that are affecting your performance.

We are here to help and we don't think less of you for asking for it!

In particular, if you feel you are routinely getting stuck with getting started on problems, make an appointment with Nick. He can sit down with you, review how you approach problems, and suggest strategies for you to try. Oftentimes, the difference between assignments taking reasonable or too much time lies in your strategies, not whether you are "smart" enough for CS. People who finish faster are usually working more effectively. We want to help you learn these techniques.

That said, **we are not on call 24/7**. Please respect that your TAs are also students with their own deadlines. **Do not approach your TAs for help outside of TA hours or labs**. Between Ed and office hours, there are plenty of chances to get help from staff who are on duty. See the email section below for the situations under which it is okay for you to email a TA directly.

Furthermore, while we are here to support you in learning CS, **finishing assignments is your responsibility, not ours**. You won't necessarily finish every assignment, and that's okay (assignments are only 50% of your course grade, and the programming components are only 35% within that, for a total of 18% of your course grade). It is up to you to turn in the best work that you can.

Ed, Email, and Contacting Staff

Please respect the following communication guidelines so that we can handle everyone's issues efficiently and effectively.

Nearly all questions related to the assignments, logistics, or materials should be posted to Ed (not sent by email). Messages sent to Ed reach the entire staff, and can get handled by whoever is next on duty. Note that it may take up to 24 hours for a post to get a response.

Routine requests (lab switch, extension request, etc) should be submitted via the corresponding form on the homepage. The forms help us track the requests and our responses so they don't get lost in our (often overflowing) inboxes. Similarly, most grading issues should start as a regrade request through Gradescope.

Problems with project partners or other logistical issues that don't involve the TAs should be emailed to the HTA list (cs0200headtas@lists.cs.brown.edu).

Messages that should be private to Nick (no HTA or TA visibility) should be sent to cs200-profs@brown.edu.

Ed

Unless a question pertains specifically to your attempted solution to a problem, please post questions and comments publically (you may keep them anonymous to other students). Questions about your code or that reveal key steps in a solution must be posted privately. The course staff reserve the right to make a private post public (but anonymous) if it asks a general question of interest to the entire class.

We will pin a post with (links to) answers to frequently-asked questions for each assignment while it is out. Before you post, please check the assignment FAQ post and do a quick Ed search to check whether your problem has already been addressed. Questions that have already been addressed in the FAQ may not get responses.

Email

You should not be emailing a UTA directly unless (a) they initiated the communication, (b) they are your lab leader and you are contacting them about a lab issue, or (c) they are your mentor and you are contacting them about a mentoring-related issue.

Routine administrative issues (missing lab attendance, adding the class late, etc) should be posted privately to Ed. Private posts are visible to the entire staff. Should your question/concern warrant more privacy, email either the HTA list (cs0200headtas@lists.cs.brown.edu) or Nick (cs200-profs@brown.edu) as appropriate.

For fastest handling, please use cs200-profs@brown.edu for all communication with Nick. This dedicated inbox helps Nick keep track of mail related to the course.

Quiet Hours and Response Times

While we check Ed every couple of hours, we aren't on all the time (especially late at night or early morning EST). **Please understand that there will usually be a delay between when you post and when we respond.** All of us have responsibilities and lives beyond CSCI0200, and the professors encourage all staff to take their non-TA commitments (including personal health!) seriously.

Nick is rarely online after 8pm EST. TAs are not permitted to respond to posts between midnight–6am EST.

Office Hours

Each TA holds at least three hours of group-style office hours, per week. Nick has two scheduled hours and is available for appointments outside of hours. Each week's schedule and location is posted on the course calendar on the website.

Contrary to what you might have experienced in high school, attending office hours does not send a message to your professors or peers that you are falling behind. On the contrary, the most successful students are usually those who come to hours early and often—whenever they have questions about topics covered in lecture or would like some guidance as they work through an assignment. Your professors like getting to know students in office hours!

We are trying a new format for office hours this year: **all office hours will use a group-style format:** simply come to the room where office hours are being held and start working—TAs will circulate around the room answering questions to help you get “unstuck” as you work. For more information, see the [Hours policy](#).

If you are stuck on some odd programming environment interaction that the TA doesn't see how to fix in a reasonable amount of time, they may tell you to post to the “Hours Referral” topic on Ed. Some TAs have more extensive experience with these environments than others, and may spot issues quicker. For tough issues, we may invite you to set up a one-on-one appointment to help diagnose the problem.

Receiving Announcements

Announcements, information about upcoming talks, interesting links, assignment clarifications (if necessary), and more will be made on Ed and announced at the start of lecture. Important messages will be pinned on Ed. Email will be used for time-critical or particularly important messages.

You are responsible for keeping up to date with pinned Ed posts and emailed announcements. In particular, you are responsible for any assignment clarification that is pinned or emailed at least 72 hours before the assignment due date.

Ergonomics

Much of the following is from <http://cs.brown.edu/about/system/ergo.html>, which contains links to other places with still more information.

Working at a computer may seem harmless but there are actually some substantial risks associated with this line of work. One source of risks is being immobile (except for the hands) for long periods of time. This can lead to back and neck and wrist injury, each of which can take a very long time to correct. Worse still, the harm done can accumulate little by little over years, only showing up much later. Now is the time to start good habits.

- Make sure you are sitting properly: Is your lower back supported? Is the top third of your screen level with your eyes? Are your feet on the floor? Are your wrists as un-bent as possible?
 - Take frequent breaks. Move the rest of your body and give your hands/wrists frequent rests. Never push it. Listen to your body: If it says you need a break, take it.
-

Research Studies

Nick may do research in how people learn computing, in collaboration with other faculty in the department who do research on computing education. This research may be used to improve this or other courses in the department, and may be published for others.

You may be offered opportunities to participate in research studies during the course. These studies are conducted by Brown students (graduate and sometimes undergraduate) working with Professor Kathi Fisler and Professor Shriram Krishnamurthi. Your participation (or not) and

performance in these studies have no effect on your course grade. Nick won't even know which students have participated until after the course has ended, as a way to avoid such influence.

Occasionally, a class-wide lab or in-class exercise could contribute data to a larger study. You will be informed about such cases if they arise. Individual students are never identified when we report on studies or their findings. If you have concerns about having your data contribute to these studies, please talk to Nick.

Please let us know if you find any mistakes, inconsistencies, or confusing language in this or any other CSCI0200 document by filling out the [anonymous feedback form](#).