



Problem Set 0: Scratch

due by 7:00pm on Fri 9/11,
per the directions at this document's end

Goals.

- Introduce some fundamental programming constructs.
- Empower you to implement your own animation, game, or interactive art.
- Impress your friends.

Academic Honesty.

All work that you do toward fulfillment of this course's expectations must be your own unless collaboration is explicitly allowed (*e.g.*, by some problem set or the final project). Viewing or copying another individual's work (even if left by a printer, stored in an executable directory, or accidentally shared in the course's virtual terminal room) or lifting material from a book, website, or other source—even in part—and presenting it as your own constitutes academic dishonesty, as does showing or giving your work, even in part, to another student.

Similarly is dual submission academic dishonesty: you may not submit the same or similar work to this course that you have submitted or will submit to another. Nor may you provide or make available solutions to problem sets to individuals who take or may take this course in the future. Moreover, submission of any work that you intend to use outside of the course (*e.g.*, for a job) must be approved by the staff.

You are welcome to discuss the course's material with others in order to better understand it. You may even discuss problem sets with classmates, but you may not share code. In other words, you may communicate with classmates in English, but you may not communicate in, say, C. If in doubt as to the appropriateness of some discussion, contact the staff.

You may even turn to the Web for instruction beyond the course's lectures and sections, for references, and for solutions to technical difficulties, but not for outright solutions to problems on problem sets or your own final project. However, failure to cite (as with comments) the origin of any code or technique that you do discover outside of the course's lectures and sections (even while respecting these constraints) and then integrate into your own work may be considered academic dishonesty.

All forms of academic dishonesty are dealt with harshly.

Getting Started.

- ☒ O hai!
- ☐ Obtain an FAS (Faculty of Arts and Sciences) Computer Account, if you haven't one already, by visiting the URL below and following the on-screen instructions.

<https://www.fas.harvard.edu/fasit/utilities/activate-pin/>

Not only will this account will provide you with access to FAS's computer facilities, it will also provide you with an email address of the form `username@fas.harvard.edu`, where username is your FAS username. Additionally, this account will allow you to access via SFTP and SSH `nice.fas.harvard.edu`, FAS's New Linux Computing Environment, use of which may be required by some problem sets. SFTP and SSH clients are available for various platforms via the course's website. Usage thereof will be taught in section and office hours.

Itching to Program?

- ☐ Surf on over to the URL below and sign up for an account on MIT's website.

<http://scratch.mit.edu/signup>

Any username (that's available) is fine, but take care to remember it and your choice of password.

- ☐ Then head to the URL below and take note of the resources available to you before you dive into Scratch itself.

<http://scratch.wik.is/Support>

You might also want to skim the tutorial below, which presents Scratch in terms a (soon-to-be) programmer should understand.

<http://www.cs.harvard.edu/~malan/scratch/>

- ☐ If not already installed on the computer you're using, download Scratch itself via the URL below.¹

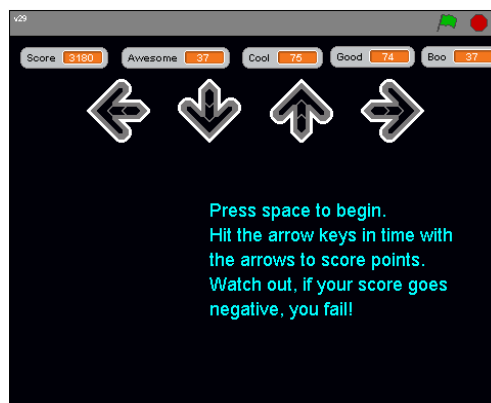
<http://scratch.mit.edu/download>

Take care to download version 1.4. Once downloaded, install the software by double-clicking its icon and/or following any on-screen directions.

- ☐ Head to

<http://www.cs50.net/lectures/>

and follow Week 0's link to **Scratch Scratch Revolution** by Ann Chi '08 (aka lindorniel). You should find yourself at MIT's website with Ann's project (screenshot below) embedded in your browser.



¹ Students running an operating system other than a supported version of Mac OS or Windows should please find a computer with a supported operating system to use for this problem set (e.g., in the Science Center, a house's lab, or a friend's room).

If you do not see Ann's project, you may need to download and install a "Java Runtime Environment" (aka "Java Virtual Machine" or "JVM"). To do so, follow the link to **Java** at <http://www.cs50.net/software/>, follow any on-screen instructions, then return to MIT's website.

Once Ann's project has loaded in your browser, press your space bar to begin!² See if you can't beat some of the high scores claimed in the page's **Comments**. Actually, email the page's URL to a roommate, and see if you can't beat him or her!

- ☐ If you've no experience (or comfort) whatsoever with programming, rest assured that Ann's project is more complex than what we expect for this first problem set. But it certainly reveals what you can do with Scratch.

Let's take a look at one other project. Return to

<http://www.cs50.net/lectures/>

and follow Week 0's link to **cherrera**, a project you haven't yet seen by Carlos Herrera. This game, too, you can play within the confines of your browser. But let's download its "source code" (*i.e.*, its .sb file). Click the link labeled **cherrera** in the page's top-right corner, just below **Download this project!** If told by a pop-up to log in, click the link labeled **Login** atop the page and log in with the username and password that you created earlier for MIT's website, then click **cherrera** again. If prompted to open or save the file (called cherrera.sb), go ahead and save it to your desktop so that it's easy to find.

Then launch the non-browser version of Scratch (*i.e.*, the one that you downloaded and installed earlier) by clicking its own icon on your desktop, in your Applications folder, or wherever you installed it. Click **File** → **Open...** toward Scratch's top-left corner, and navigate your way to Carlos's project, wherever it ended up on your hard drive.

Spend some time looking over Carlos's scripts. Don't forget that each sprite has its own set of scripts. Try to get a sense of how the overall program works. Try making some changes, even while the program is running, to see how the program responds. Note that this project is probably a bit simpler than we expect of you for this problem set, but it's a good one to learn from because it's pretty easy to follow. And do appreciate that this game, like all Scratch projects, reduces quite literally to some basic building blocks.

- ☐ Feel free to download the source code for a few more projects, either from

<http://scratch.mit.edu/galleries/>

or from Week 0 at

<http://www.cs50.net/lectures/>

even if you already saw it in lecture.

² If the game doesn't start, try clicking the green flag in the project's top-right corner, then hit your space bar again.

Or pull up some of the projects that come with Scratch by clicking **File** → **Open...** toward Scratch's top-left corner followed by the icon labeled **Examples** in the window that ultimately appears.

For each program, run it to see how it works overall and then look over its scripts to understand how it works underneath the hood. Feel free to make changes to scripts and observe the effects. Once you can say to yourself, "Okay, I think I get this," you're ready to proceed.

- ☐ Now your own adventure begins. Your task for this problem is, quite simply, to have fun with Scratch and implement a project of your choice (be it an animation, a game, interactive art, or anything else), subject only to the following requirements.
 - i. Your project's filename must be username.sb, where username is your FAS (not MIT) username in all lowercase.
 - ii. Your project must have at least two sprites, at least one of which must resemble something other than a cat.
 - iii. Your project must have at least three scripts total (*i.e.*, not necessarily three per sprite).
 - iv. Your project must use at least one condition, one loop, and one variable.
 - v. Your project must use at least one sound.
 - vi. Your project should be more complex than most of those demonstrated in lecture (many of which, though instructive, were quite short) but it can be less complex than, say, Oscartime, Raining Men, and Scratch Scratch Revolution. As such, your project should probably use a few dozen puzzle pieces overall.

Feel free to peruse additional projects online or those that come with Scratch for inspiration, but your own project should not be terribly similar to any of them. Try to think of an idea on your own, and then set out to implement it. If, along the way, you find it too difficult to implement some feature, try not to fret; alter your design or work around the problem. If you set out to implement an idea that you find fun, you should not find it hard to satisfy this problem set's requirements. If you suspect your program might fall short of our expectations, feel free to ask a member of the staff for an opinion prior to submitting.

Alright, off you go. Make us proud! If you have questions or want a hand making your project even better, do take advantage of office hours, the schedule for which can be found at <http://www.cs50.net/ohs/>. Alternatively, drop us a note at help@cs50.net.

- ☐ Once finished with your project, upload it to your account on MIT's server by clicking **Share** → **Share This Project Online...** toward the top of Scratch's window. Provide "Your Scratch website login name" (*i.e.*, username) and password that you chose earlier, along with a name for your project, some notes (*e.g.*, some instructions or a description), and zero or more tags. Best to leave **Compress sounds and images** checked. Then click the **OK** button. Assuming you're informed that "Your project is now online at scratch.mit.edu," head on over to the URL below.³

<http://scratch.mit.edu/login>

³ If informed that your project is too large to be uploaded, try to decrease its size by clicking **Edit** → **Compress Sounds...** and/or **Edit** → **Compress Images...** atop Scratch's window. You may need to experiment with different levels of compression.

Log in (if you aren't still logged in) with your username and password. Once logged in, click **my stuff** toward the page's top-right corner. You should see the project you just uploaded among **My Projects**. Go ahead and click its name or icon. Your project should be embedded (and start playing) in the window that appears. Take note of the URL in your browser's address bar. That's your project's URL on MIT's website, and you'll need to know it later.

- ☐ When done admiring your work, head on over to the URL below.

<http://www.cs50.net/galleries/2009/fall/scratch/>

Make sure that you're still logged into MIT's website. (If you see "Welcome" followed by your username atop the page, you are.) If not, click **Login** to log in again.

Toward the page's right-hand side, click **add my projects** and wait for a window to appear. Once that window appears, click your own project's name, wait for a checkmark to appear to in the box to the left of it, then click **Accept**. If you return to the URL above, you should find that your project has been added to Fall 2009's gallery for others to enjoy. If not, try once more or contact the staff for assistance.

If you really, really don't want your work included in the course's gallery, you may opt out of this particular step. But you must still upload your project to MIT's server (by clicking **Share** → **Share This Project Online...**), per the previous step. And you must still "submit" your project to us, per the next step.

Submitting Your Work.

- ☐ When you're ready to "submit" your project to us, surf on over to the URL below to answer some questions.

<http://www.cs50.net/surveys/pset0/>

Be extra-sure that your FAS username and your project's URL are correct, else we may overlook your submission! Then click **Done** to "submit" this, your first CS 50 problem set!

- ☐ kthxbai