

Course Overview

21M.385 / 21M.585 / 6.4550 Interactive Music Systems

- Term: Fall 2024
- Meets:
 - MW 11:00-12:30, alternating lectures and sections (see [Syllabus \(https://canvas.mit.edu/courses/24968/assignments/syllabus\)](https://canvas.mit.edu/courses/24968/assignments/syllabus).)
 - Lectures: 4-270
 - Sections: 4-270 (Section 1) or 4-152 (Section 2)
 - Hack hours / occasional workshops: Friday 4-6 pm in 4-152
- Instructor: Eran Egozy (egozy@mit.edu (<mailto:egozy@mit.edu>))
 - Office hours: Monday 1:30-2:30 pm in 24-033G, Thursday 3-4pm [on zoom \(https://mit.zoom.us/my/egozy\)](https://mit.zoom.us/my/egozy), or by appointment.
- TA: Alex Studer (astuder@mit.edu (<mailto:astuder@mit.edu>))
 - Office hours: Tuesday 3-4 pm in 24-033G; Friday 4-5 pm in 24-033G or [on zoom ↗ \(https://mit.zoom.us/my/alex.studer\)](https://mit.zoom.us/my/alex.studer)

See the [Syllabus \(https://canvas.mit.edu/courses/24968/assignments/syllabus\)](https://canvas.mit.edu/courses/24968/assignments/syllabus) for the course schedule and deadlines.

Description

Interactive Music Systems is a hands-on programming and design course that explores audio synthesis, musical structure, HCI (human computer interaction), and visual presentation as the ingredients for the creation of engaging real-time interactive musical experiences.

These experiences allow users to connect with music more deeply than through passive listening. The most successful ones (such as Harmonix's *Guitar Hero*, *Rock Band*, *Fantasia: Music Evolved*) give users intuitive control, greater musical insight, and a deeper emotional response to music.

Students will learn about the principles, design considerations, and aesthetic qualities of interactive music systems by exploring the following topics:

- Music perception and audio synthesis
- Dynamic multi-track audio mixing and looping
- Programmatic MIDI and event sequencing
- Generative composition systems including rhythmic and melodic synthesis
- Exploration of non-standard control devices such as game controllers, motion sensors (Kinect, Leap Motion), and pad controllers.
- Analysis and application of design elements in music games.
- Building graphics for UI, music visualization, and aesthetic cohesion.

Prerequisites: 21M.301, 6.101.

Class Structure

We alternate between Lecture and Section meetings and also offer occasional optional workshops.

Lectures consist of conceptual presentation and live coding to demonstrate how concepts are put into practice.

Sections consist of student presentations and in-class coding exercises (labs) related to the week's material. You will be assigned to a section in the first week of class.

Workshops are optional TA-run sessions on occasional Friday afternoons to cover additional material and to provide additional support on assignments.

Assignments

Problem sets are coding assignments that have an engineering component and a [creative component \(https://canvas.mit.edu/courses/24968/pages/pset-creative-guidance\)](https://canvas.mit.edu/courses/24968/pages/pset-creative-guidance). Psets are due at 11:59pm - see the [syllabus \(https://canvas.mit.edu/courses/24968/assignments/syllabus\)](https://canvas.mit.edu/courses/24968/assignments/syllabus) for deadlines. You may work on assignments alone or with a partner from class. You can use [pset partners \(https://psetpartners.mit.edu\)](https://psetpartners.mit.edu) to find a buddy if you'd like. If you work together, you should accompany your pset submission with

a note describing how you worked together. Assignments should be uploaded to Canvas. Each student must upload their own assignment. Each new assignment builds on the previous ones.

Pset demos take place during the section after each pset is due where three students demonstrate the creative parts of their solution. Each student is required to present a pset once during the semester.

Labs take place during sections. They are graded pass/fail and must be uploaded to Canvas by the end of the day. Lab solutions are posted soon after.

Exploration presentations are short presentations given in sections, for which you'll discover, analyze, and present an interesting new media work related to music technology, interactive music, generative music, or music visualization. Read more info at the [Exploration Presentations Page \(https://canvas.mit.edu/courses/24968/pages/exploration-presentations-home\)](https://canvas.mit.edu/courses/24968/pages/exploration-presentations-home).

The **Graduate assignment**, [Exploring NIME \(https://canvas.mit.edu/courses/24968/assignments/316722\)](https://canvas.mit.edu/courses/24968/assignments/316722), must be completed by students taking 21M.585, the graduate version of this subject.

We use the Python programming language and the [Kivy \(https://kivy.org/\)](https://kivy.org/) application framework for all classwork. You will also need a few freely available software packages. Read the class [Installation Instructions \(https://canvas.mit.edu/courses/24968/pages/installation-instructions\)](https://canvas.mit.edu/courses/24968/pages/installation-instructions).

For pset 5, we use a hardware input device, the [Leap Motion Controller \(https://www.ultraleap.com/product/leap-motion-controller/\)](https://www.ultraleap.com/product/leap-motion-controller/). Controllers are loaned out from the Lewis Music Library.

Final Project

This class has a Final Project lasting the final six weeks of the semester. Final projects are done in teams of three students per team. During this time, teams will meet individually with the instructors in lieu of regular lectures.

Each team is required to make three presentations:

- A project proposal and initial milestone presentation.
- A second interim milestone presentation.
- A final presentation at the end of the semester (which will be recorded).

The final class of the semester will be a demo party where we all get to try out each other's final projects. You can view projects from previous semester at the [bottom of the IMS webpage \(http://musictech.mit.edu/ims\)](http://musictech.mit.edu/ims).

Final project teams may elect to use hardware devices as well. Logistics (buying or borrowing) must be discussed with the instructor ahead of time.

Attendance and Support

IMS is a highly interactive class. We expect students to be fully engaged during lectures and peer presentations, and to practice new skills in labs. There is no required text and relatively little required reading. All topics are conveyed during lectures and sections. While lecture notes are available, they are only meant to serve as a reminder and do not cover all the material presented in lectures.

Attendance, therefore, is mandatory and is a part of your grade.

That said, stuff happens. If you need to miss class due to an extenuating circumstance, email us (instructor and TA) in advance and you may be excused in part or in full on a case-by-case basis. You are granted accommodation for missing class due to religious / faith practice, but please let us know in advance. If you don't feel comfortable coming to one of us, you may [contact S3 for an excuse note \(https://studentlife.mit.edu/s3/support-advocacy/excuse-notes\)](https://studentlife.mit.edu/s3/support-advocacy/excuse-notes).

We will endeavor to record and post lecture videos to help you catch up if you miss a class. You should also contact classmates to find out what you missed.

And as always, please contact S3 for any issues (medical, personal, or otherwise) that hinder your ability to focus on academics. You can reach out to a dean you have worked with in the past, join their [virtual help queue \(https://s3cp-s3.mit.edu/queue\)](https://s3cp-s3.mit.edu/queue), or e-mail [s3-support@mit.edu \(mailto:s3-support@mit.edu\)](mailto:s3-support@mit.edu).

Grading

Grading varies slightly depending on if you are enrolled in 21M.385 (undergrad version) or 21M.585 (graduate version).

	Undergraduate	Graduate
--	----------------------	-----------------

Attendance	10%	5%
Pset Demo and EP	5%	5%
Labs	5%	5%
Homework (psets)	45%	40%
Graduate Assignment	N/A	10%
Final Project	35%	35%

Grade cutoffs are the standard set with [97, 100] = A+, [93, 97] = A, [90, 93] = A-, [87, 90] = B+, etc...

There is no final exam.

Deadlines and Extensions

Late psets are penalized 10/24% per hour late (i.e., 10% per day), with a maximum penalty of 50%.

Every student is allowed one pset extension, which must be requested before the deadline using the link on the course [Home Page](#) (<https://canvas.mit.edu/courses/24968/pages/home-page>). When you request an extension, you must propose a new, reasonable deadline. A reasonable deadline is one that you feel confident hitting given your overall workload, but that won't put you too far behind in the class.

We don't accept late labs unless you have an extenuating circumstance, in which case please email the TA.

Academic Integrity and Collaboration

Read MIT's [Academic Integrity Policy](#) (<http://integrity.mit.edu>). It applies here. In particular, you will be coding for this class, so read the [Writing Code](#) (<http://integrity.mit.edu/handbook/writing-code>) section. The purpose of this class is to teach you how to build interactive music systems yourself. If you cheat, you will limit your own understanding and abilities and put both of us in a difficult situation. Let's avoid that. If you're struggling with the material, reach out for support. We're here to help you learn.

Working together is encouraged. For psets, you can share ideas, discuss strategies, and help each other debug code. However, each person is responsible for implementing their own system and writing their own code.

In the final project, the entire team works on a single codebase. It is not necessary for each person to implement their own separate system. You may code as a group, pair-program, or collaborate in any way you see fit. Each team member must contribute an equal share of the work.

You may look online for Python implementation tips and ideas when doing assignments. Stack Overflow is a great resource, for example. Cite the source in a code comment. If you do copy, limit yourself to a few lines of code. Do not copy or use whole modules or large functions.

Do not use ChatGPT, GitHub Copilot, or any other code generators. This is considered cheating and often produces incorrect answers anyway.

You are allowed to copy, use, and modify code provided to you by the instructors, or code that you wrote for previous assignments. In fact, many assignments assume you have access to working code from previous assignments.

Do not post any code we distribute to you online. If, for example, you want to use GitHub to manage your course materials, make sure to do so in a private repo.

For the final project, your design may call for using larger amounts of code than is permitted for assignments. The specific nature of external code usage for the final project must be discussed with the instructors.

Disability and Access Services

MIT is committed to the principle of equal access. Students who need disability accommodations are encouraged to speak with Disability and Access Services (DAS), prior to or early in the semester so that accommodation requests can be evaluated and addressed in a timely fashion. If you have a disability and are not planning to use accommodations, it is still recommended that you meet with DAS staff to familiarize yourself with their services and resources. Please visit the [DAS website](https://studentlife.mit.edu/das) (<https://studentlife.mit.edu/das>) for contact information.

If you are approved for accommodations, we're ready to assist with implementation - just let us know.

Course Syllabus

[Jump to Today](#)

Interactive Music Systems: Spring 2024

Meets: MW 11 - 12:30

Lectures: 4-270

Sections: 4-270 and 4-152

Read the [Course Overview \(https://canvas.mit.edu/courses/24968/pages/course-overview\)](https://canvas.mit.edu/courses/24968/pages/course-overview) for a description of the course and class expectations.

Schedule

Class Schedule

Week of	Sun	Mon	Tue	Wed	Fri
Feb 4		Lec: Intro / Overview		Lec: Audio & sine synthesis	installation, numpy workshop; setup & signups due
Feb 11		Lec: Kivy framework		Sec: EP, lab 1	
Feb 18		<i>Holiday (Pres Day)</i>	Lec: Wave files & audio buffers; pset 1 due	Sec: EP, pset 1 demos, lab 2	
Feb 25		Lec: Graphics	pset 2 due	Sec: EP, pset 2 demos, lab 3	graphics workshop (4-261)
Mar 3		Lec: MIDI, fluidsynth, scheduler	pset 3 due	Sec: EP, pset 3 demos, lab 4	
Mar 10		Lec: Input devices, leap motion	pset 4 due	Sec: EP, pset 4 demos, lab 5	
Mar 17		Lec: FP overview, music games	pset 5 due	Sec: EP, pset 5 demos, lab 6	
Mar 24		<i>Holiday (Spring Break)</i>		<i>Holiday (Spring Break)</i>	
Mar 31		Lec: EP, FP process, idea pitches, Fantasia: Music Evolved	pset 6 due; team prefs due	Sec: EP, pset 6 demos. Lec: Guest Speaker Juan Bello	FP teams announced
Apr 7	FP proposal due	Team meetings (zoom)	NIME video due	Lec: EP, Guest Speaker Jordan Rudess	FP workshop (4-253);
Apr 14	FP log 1 due	<i>Holiday (Patriot's Day)</i>		Milestone 1 presentation	pset 7 due
Apr 21	FP log 2 due	Working session		Team meetings (24-033G)	
Apr 28	FP log 3 due	Milestone 2 presentation		Play testing	
May 5	FP log 4 due	Final presentation (A)		Final presentation (B)	
May 12		IMS Party!			

Color coding:

- **Assignment due**
- Assignment due (21M.585 only)
- Sections meeting (split between 4-270 and 4-152)