

Fall 9-1-2017

MART 220.01: Creative Coding II

Michael A. Musick

University of Montana, Missoula

Let us know how access to this document benefits you.

Follow this and additional works at: <https://scholarworks.umt.edu/syllabi>

Recommended Citation

Musick, Michael A., "MART 220.01: Creative Coding II" (2017). *Syllabi*. 8756.
<https://scholarworks.umt.edu/syllabi/8756>

This Syllabus is brought to you for free and open access by the Course Syllabi at ScholarWorks at University of Montana. It has been accepted for inclusion in Syllabi by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact scholarworks@mso.umt.edu.

Syllabus I Creative Coding 2, MART 220

Date modified: 2017-09-05

Table of Contents

- [Course Overview](#)
 - [Professor/Instructor](#)
 - [Course Websites](#)
 - [Course Requirements](#)
 - [Technology, Tools, and Required Software](#)
 - [Books & Supplies](#)
- [Policies](#)
 - [Course Evaluation](#)
 - [Grades](#)
 - [Final Grades](#)
 - [Assignment Grades](#)
 - [Plagiarism & Cheating Policy](#)
 - [Students with Disabilities](#)
 - [Changes to the Course](#)

COURSE OVERVIEW

This course is designed to teach code so that it can become a tool for art and creativity. This course furthers this goal by picking up where Creative Coding 1 (MART 120) leaves off.

The goal is to begin a journey in which the creative and technical possibilities of the computer become a tool in your palate. These tools will become an extension of you, something you think through, express yourself through, and can contribute to the world through. Ultimately, this course is designed to offer you a deeper understanding of the essential possibilities of computation.

This course may look at:

- nature
- physical computing
- natural language processing
- sound/music
- video
- external libraries
- and more

The course will conclude, with your design, development, and implementation of a final project of your choosing.

Professor/Instructor

- [Professor Michael Musick](#)
- E-Mail: michael.musick@umontana.edu.
- Office: McGill Hall, 232.

Course Websites

- [Course GitHub Repo](#) (This git repository holds code examples, an issues board, as well as course Wiki.)
- [Direct Link to Course Wiki](#)
- [mart220 Slack Channel](#)
 - Slack is a communication tool used in industry, startups, and art teams. We will be trying it this semester as our primary tool of communication for the class. You are **REQUIRED** to participate in Slack and check it regularly. (I would encourage you to download the Slack app for your computer and mobile devices.)
 - [Link to Join](#)

Course Requirements

Time

You should plan to spend 8-15 hours of work a week on homework and coding outside of class time (in this case, the time it takes you to read/watch the weeks content). **This is a very intense course that will require a significant time commitment on your part.**

This means you will need excellent time management skills. Schedule time for this course in your week, or you will not be capable of being successful.

Creativity

You are taking this course because it offers the opportunity to learn coding skills through creative means. Please take the time to be creative in your weekly code assignments. Pushing the boundary creatively will make this course more meaningful.

Cleanliness

Coding is hard. But, messy code, documents, and files will make it more difficult. Try to be as clean and tidy in your organization for this class. When writing code, try to keep it organized, and provide many comments. This will make it easier for the grader, the instructor, and you.

Technology, tools, & required software

- **p5.js**
 - The Javascript-based p5.js will be used as the primary development environment.
- **Processing**
 - A Java-based environment for creative coding.
- **Arduino**
 - A microcontroller for physical computing and sensing.
- **Python**
 - A hugely popular language for general and scientific computing.
- **Markdown**
 - All blog and response entries for each homework will be written using the markdown language. This is a simple, light-weight markup language that you will learn in week 2.
- **Editor**
 - You should install a text editor. If you do not already have a text editor installed I suggest GitHub's [Atom](#).
- **Git**
 - You will need to use Git; a free and open source distributed version control system. This should already be installed on your computer.
 - You can work with git via the terminal or [command line](#).
 - Although it is recommended you use a GUI based front-end application. Such as;
 - [GitHub Desktop](#)
 - [SourceTree](#)
- **GitHub**
 - Additionally, you will be expected to maintain a [github.io](#) repository and website via [GitHub](#) where you will post your weekly homework assignments.
- **Browser**
 - You should ensure you have a modern browser installed on your computer, such as Safari, Google Chrome, or Firefox.

Books & Supplies

Required

The following book is required.

- [Shiffman, Daniel. "The Nature of Code." Self-Published, 2012.](#) (Note: This book is freely available online from the provided link. i.e. You do not need to purchase if you are fine with reading via the browser.)
- An Arduino "Experimentation Kit". (Roughly \$85-\$90) I suggest;
 - [Adafruit ARDX - v1.3](#).
 - Or, [SparkFun Inventor's Kit V3.3](#)
 - You will need an;
 - Arduino
 - LED's (20 or so of multiple colors)
 - Toy DC Motor
 - Servo Motor
 - Push Buttons
 - Potentiometers
 - Photo Resistor
 - Transistors
 - Resistors
 - Diodes

Optional

The following are additional optional resources which may prove useful.

- [Learning Processing](#) is another useful resource for beginning creative coders.
- If you are interested in some supplemental reading on HTML and CSS, The [HTML & CSS book](#) is a nice one.

Policies

Course Evaluation

Students work and progress will be assessed through;

- Homework Assignments
 - These typically will be code-based assignments
 - Code-based assignments will always include a reflection report
 - Weekly assignments may also include or instead be problem sheets that will be completed via Moodle.
- Class participation via group problem solving, support, and forum discussions.

Homework

We will have weekly assignments that are relevant to material from recent content. These assignments are required and you will need to talk about them in your technical reports. It is expected that everyone in the class will create and maintain a github repository for their assignments.

The assignments are your opportunity to put your skills to the test. They're designed to test and deepen your knowledge while giving me valuable information about how you're progressing in the class. **Finished is better than perfect.** An attempt at them is better than nothing at all.

In addition to the weekly code assignments, you will be expected to also complete a weekly reflection report. In this, you will discuss;

- your weeks progress,
- what did & did not work,
- comments on any frustrations you had with the language,
- and critically analysis/evaluation of how much time was spent learning syntax, structure, & programming concepts vs. actually programming.

Homework will be submitted both via Moodle and the [class GitHub Homework wiki](#). You will submit links to your code, sketch, and technical report (both hosted via your GitHub repository) for every coding assignment to the Wiki. Exams, quizzes, question sheets, and reports will be submitted via Moodle. Every homework assignment will have an entry in the Moodle Course Page.

Due to the size of the class and the nature of grading code-based homework, you may not receive individual feedback for assignments. Please feel free to send specific queries about homework or reasons for why you received the grade you did to the instructor via email.

Grades for all assignments will be based primarily on the student's ability to:

1. Demonstrate an understanding of the specific characteristics and integrative capabilities of the assigned topic in your own words and code.
2. Articulate a clear and concise perspective. Cutting and pasting or copying word for word off the Internet will result in loss of points.
3. Present an organized file/program, as well as technical report; including proper and punctual delivery of the assignment files.
4. Demonstrate creativity beyond the expected technical requirements.

IMPORTANT: Assignments handed in after the due date and time will have points deducted for lateness. This will be in addition to any points deducted for content. Those that are uploaded late but within one week of the due date will lose 5% for lateness. For those uploaded after that, the number of deducted points will be at the discretion of the faculty.

Participation

This class will be participatory, you are expected to participate in discussions and give feedback to other students through on-line participation with their projects.

Final Project

This course will culminate with final projects. You are expected to push your abilities to produce something that utilizes what you have learned in the class that is useful in some manner to yourself or the world.

The final project will be a self-developed creative work utilizing the principles and ideas developed throughout this course. This project should equal roughly the same amount of work as 3 regular homework assignments. In other words, this is a significant creative piece.

Required: Please note, the final project is required. Failure to complete a final project will result in a letter grade of F for the course. This is regardless of the students current grade standing in the course. There are no exceptions to this policy.

Grades

Final Grades

Grades will be determined according to the following breakdown:

- Regular Assignments 60%
- Final Project 30%
- Participation 10%

Letters are assigned according to the following final course percentages:

Grade	% Range
A	93-100
A-	90-93
B+	87-90
B	83-87
B-	80-83
C+	77-80
C	73-77
C-	70-73
D	60-70
F	0-59

Assignment Grades

For each project your grade will be assessed upon the following:

- Principles
- Creativity/Thoughtfulness
- Craft.

Below gives you a sense of the spectrum, giving examples for high, middle and low grades.

Please note:

A letter grade of "C" is an average grade. Average is not bad. It means average. Most of us are average, in some way. An average level of work is being done. It does not mean perfection. It means that the work still has room to develop at it's current state. It also means that the work has noticeable achievements and evidence of some acquisition of knowledge. I look at this on an individual basis as well as across the whole class.

Please read examples of each assessment point below to get an idea of my expectations. Note that a "B" is between Satisfactory and Superior.

- **Principles:**
 - Superior - A - Your work shows evidence and understanding of programming concepts discussed in readings, lectures, and exercises, as you bend them to your will.
 - Satisfactory - C - Your work shows evidence of concepts and is still developing an understanding of course material. You understand some aspects, but aren't utilizing them fully. Usage is cursory. Keep pushing your work and review the material to revisit how you can integrate it to your work.
 - Unacceptable - D/F - Your work shows some evidence of concepts discussed, but lacks key understanding, confidence, robustness and authority. Aspects are lacking. Reviewing course material is required. Ask questions in class. Manage your time better.
- **Creativity, Thoughtfulness:**
 - Superior - A - Your work demonstrates your personality and a great depth of engagement with the material. It's extremely evident that you are thinking, exploring, playing and taking risks. You are creating wonderful experiences.
 - Satisfactory - C - You are executing your ideas, but more time is needed to consider more deeply about what is conceptually and physically happening. They are barely getting off the ground. Or have large bits that are broken. Do more research and exploring. Play.
 - Unacceptable - D/F - Your work is so straightforward that it's flat. It's barely coded (or badly broken). It's copied from elsewhere and not expanded upon. Question and Iterate your work to push your it further. Read. Play. Get off the lame track and get inspired.
- **Craft:**
 - Superior - A - Your work shows delicate care and consideration to presentation and professionalism. You code is neat, clean, commented and structured. Your friends consider you "Type A." Your style is evident.
 - Satisfactory - C - Your work shows the birth of your ideas, but further time and iteration can really push your work to excel. Your code is there, but messy. It can be simplified and made cleaner. What you put into it, is what you get out of it. Practice makes perfect!
 - Unacceptable - D/F - Your work is rushed and looks like it was done on your train-ride in or the night before. Make your work something you are proud of. You are here to build your portfolio after all, aren't you?

Plagiarism & Cheating Policy

Students are expected to adhere to academic conduct policies of the University of Montana as explained in the [Student Conduct Code](#). All students should act with personal integrity, respect other students' dignity, rights, and property, and help create and maintain an environment in which all can succeed.

Dishonesty will not be tolerated in this course. This includes, but is not limited to, cheating on tests, cheating on assignments, fabricating information or citations, having unauthorizd possession of examinations, submitting work of another person or work previously unused, or tampering with the academic work of other students.

Plagiarism is the presentation of the work of another without acknowledgement. As defined by the [University of Montana's Student Conduct Code](#), plagiarism is "Representing another person's words, ideas, data, or materials as one's own." Students may use information and ideas expressed by others, but this use must be identified by appropriate referencing.

Students who cheat or plagiarize will receive the academic sanctions, which may include an "F" grade on the assignment, examination, and/or in the course. Students will also be reported to the Dean of Students for possible further disciplinary action.

Using Code Found Elsewhere

It's super easy to find code online. If you use code from elsewhere (which you will at times), I expect you to site the work and author, as well as to comment each line, as to what each line does programmatically. Do *not* summarize several lines of code from a high level (ie, TV Guide). I expect you to comment each line on a granular level. In addition in these cases, I am also looking for significant modification of the code, for you to enact your own ideas and to experiment heavily.

Significant modification means beyond variable name and value changes. It's bending these concepts to your idea, especially graphically. It's not a copy and paste job.

Also, never more than 40% of your code may be supplied from elsewhere. Period.

If you use code from online, whether for inspiration, modification or reference, I expect to see a link in your comments from where you got the code and who wrote it. Otherwise it will be considered as plagiarism, and you will fail the assignment. The code must have a reference, along with URL and be commented out LINE BY LINE.

Students with Disabilities

The University of Montana assures equal access to instruction through collaboration between students with disabilities, instructors, and Disability Services for Students. Students with disabilities are encouraged to plan ahead and can contact [Disability Services for Students \(DSS\)](#). If you think you may have a disability adversely affecting your academic performance, and you have not already registered with Disability Services, please contact Disability Services in Lommasson Center 154, or call (406)243-2243. I will work with you and Disability Services to provide an appropriate modification.

Changes to the Course

I reserve the right to change the intended content of this course throughout the semester. This may be done to adjust for the speed of the class, to better meet educational goals, or to account for changes in technology.