ASU

ARIZONA STATE UNIVERSITY

GENERAL STUDIES COURSE PROPOSAL COVER FORM

Course information:

*Copy and paste **current** course information from Class Search/Course Catalog.*

Academic Unit Arts, Media and Engineering Department Herberger Institute

Subject AME Number 111 Title Introduction to Digital Culture Units: 3

Is this a cross-listed course? (Choose one)
If yes, please identify course(s) N/A

Is this a shared course? (choose one) If so, list all academic units offering this course N/A
Course description:

Requested designation: (Choose One)

*Note- a **separate** proposal is required for each designation requested*

Eligibility:

Permanent numbered courses must have completed the university's review and approval process.
For the rules governing approval of omnibus courses, contact the General Studies Program Office at (480) 965–0739.

Area(s) proposed course will serve:

A single course may be proposed for more than one core or awareness area. A course may satisfy a core area requirement and more than one awareness area requirements concurrently, but may not satisfy requirements in two core areas simultaneously, even if approved for those areas. With departmental consent, an approved General Studies course may be counted toward both the General Studies requirement and the major program of study.

Checklists for general studies designations:

Complete and attach the appropriate checklist

- Literacy and Critical Inquiry core courses (L)
- Mathematics core courses (MA)
- Computer/statistics/quantitative applications core courses (CS)
- Humanities, Fine Arts and Design core courses (HU)
- Social and Behavioral Sciences core courses (SB)
- Natural Sciences core courses (SQ/SG)
- Global Awareness courses (G)
- Historical Awareness courses (H)
- Cultural Diversity in the United States courses (C)

A complete proposal should include:

☒ Signed General Studies Program Course Proposal Cover Form
☒ Criteria Checklist for the area
☒ Course Syllabus
☒ Table of Contents from the textbook, and/or lists of course materials

Contact information:

Name Casey Kazmierczak Phone 480-727-2849

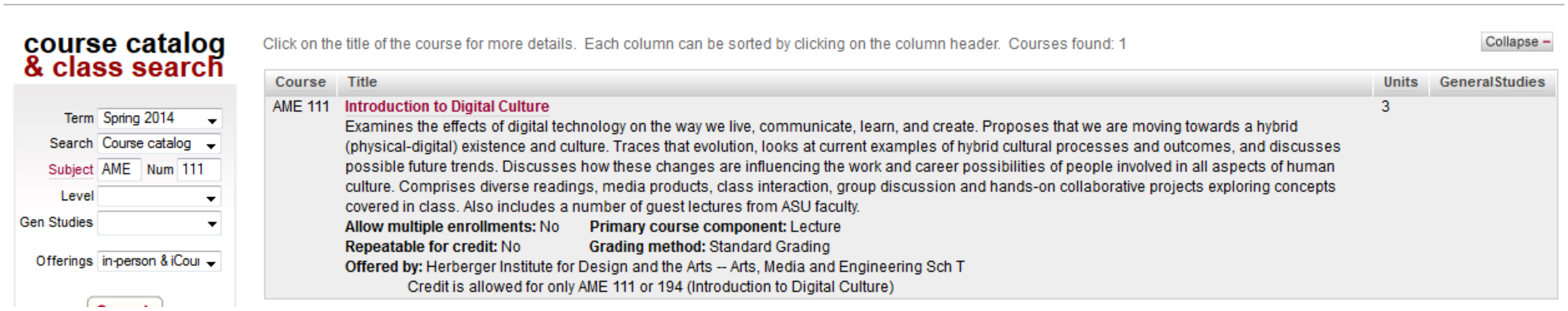
Mail code 5802 E-mail: casey.kaz@asu.edu

Department Chair/Director approval: *(Required)*

Chair/Director name (Typed): Xin Wei Sha Date: 1/7/13

Chair/Director (Signature):

Rev. 1/94, 4/95, 7/98, 4/00, 1/02, 10/08, 11/11/ 12/11, 7/12



Term Spring 2014

Search Course catalog

Subject AME Num 111

Level

Gen Studies

Offerings in-person & iCour

Click on the title of the course for more details. Each column can be sorted by clicking on the column header. Courses found: 1

Course	Title	Units	GeneralStudies
AME 111	**Introduction to Digital Culture** Examines the effects of digital technology on the way we live, communicate, learn, and create. Proposes that we are moving towards a hybrid (physical-digital) existence and culture. Traces that evolution, looks at current examples of hybrid cultural processes and outcomes, and discusses possible future trends. Discusses how these changes are influencing the work and career possibilities of people involved in all aspects of human culture. Comprises diverse readings, media products, class interaction, group discussion and hands-on collaborative projects exploring concepts covered in class. Also includes a number of guest lectures from ASU faculty. **Allow multiple enrollments:** No **Primary course component:** Lecture **Repeatable for credit:** No **Grading method:** Standard Grading **Offered by:** Herberger Institute for Design and the Arts -- Arts, Media and Engineering Sch T Credit is allowed for only AME 111 or 194 (Introduction to Digital Culture)	3	

Arizona State University Criteria Checklist for

MATHEMATICAL STUDIES [CS]

Rationale and Objectives

The **Mathematical Studies** requirement is intended to ensure that students have skill in basic mathematics, can use mathematical analysis in their chosen fields, and can understand how computers can make mathematical analysis more powerful and efficient. The **Mathematical Studies** requirement is completed by satisfying both the **Mathematics [MA]** requirement and the **Computer/Statistics/Quantitative Applications [CS]** requirement explained below.

The **Mathematics [MA]** requirement, which ensures the acquisition of essential skill in basic mathematics, requires the student to complete a course in College Mathematics, College Algebra, or Precalculus, or demonstrate a higher level of skill by completing a mathematics course for which any of the first three courses in a prerequisite.

The **Computer/Statistics/Quantitative Applications [CS]** requirement, which ensures skill in real world problem solving and analysis, requires the student to complete a course that uses some combination of computers, statistics, and mathematics.

Approved: Feb. 2000

Proposer: Please complete the following section and attach appropriate documentation.

ASU--[CS] CRITERIA

A COMPUTER/STATISTICS/QUANTITATIVE APPLICATIONS [CS] COURSE MUST SATISFY ONE OF THE FOLLOWING CRITERIA: 1, 2, OR 3

YES	NO		Identify Documentation Submitted
		1. Computer applications*: courses must satisfy both **a** and **b**:	
☒	□	**a.** Course involves the use of computer programming languages or software programs for quantitative analysis, modeling, simulation, animation, or statistics.	1. Syllabus 2. Course Topic Outline 3. Final Project Student Handout
		b. Course requires students to analyze and implement procedures that are applicable to at least one of the following problem domains **(check those applicable)**:	
□	☒	**i.** Spreadsheet analysis, systems analysis and design, and decision support systems.	
☒	□	**ii.** Graphic/artistic design using computers.	1. Motion Capture Student Assignment 2. Assigned Readings "Overview of Processing" and "Getting Started with Processing." 3. Course Topic Outline
☒	□	**iii.** Music design using computer software.	1. Assigned MaxMSP readings 2. Final Project Student Handout 3. Course Topic Outline 4. Lecture Script from Week 2
□	☒	**iv.** Modeling, making extensive use of computer simulation.	
□	☒	**v.** Statistics studies stressing the use of computer software.	

ASU--[CS] CRITERIA

*The **computer applications** requirement **cannot** be satisfied by a course, the content of which is restricted primarily to word processing or report preparation skills; learning a computer language or a computer software package; or the study of the social impact of computers. Courses that emphasize the use of a computer software package or the learning of a computer programming language are acceptable, provided that students are required to understand, at an appropriate level, **the theoretical principles embodied in the operation of the software and are required to construct, test, and implement procedures that use the software to accomplish tasks in the applicable problem domains.**

		2. Statistical applications: courses must satisfy both **a** and **b**.	
□	☒	**a.** Course has a minimum mathematical prerequisite of College Mathematics, College Algebra, or Precalculus, or a course already approved as satisfying the MA requirement.	
□	☒	**b.** The course must be focused principally on developing knowledge in statistical inference and include coverage of all of the following:	
YES	**NO**		**Identify Documentation Submitted**
□	☒	**i.** Design of a statistical study.	
□	☒	**ii.** Summarization and interpretation of data.	
□	☒	**iii.** Methods of sampling.	
□	☒	**iv.** Standard probability models.	
□	☒	**v.** Statistical estimation	
□	☒	**vi.** Hypothesis testing.	
□	☒	**vii.** Regression or correlation analysis.	
		3. Quantitative applications: courses must satisfy both **a** and **b**.	
□	☒	**a.** Course has a minimum mathematical prerequisite of College Mathematics, College Algebra, or Precalculus, or a course already approved as satisfying the MA requirement.	

ASU--[CS] CRITERIA

		b. The course must be focused principally on the use of mathematical models in quantitative analysis and decision making. Examples of such models are:	
□	☒	**i.** Linear programming.	
□	☒	**ii.** Goal programming.	
□	☒	**iii.** Integer programming.	
YES	**NO**		**Identify Documentation Submitted**
□	☒	**iv.** Inventory models.	
□	☒	**v.** Decision theory.	
□	☒	**vi.** Simulation and Monte Carlo methods.	
□	☒	**vii.** Other (explanation must be attached)	

Course Prefix	**Number**	**Title**	**Designation**
AME	111	Introduction to Digital Culture	CS

Explain in detail which student activities correspond to the specific designation criteria. Please use the following organizer to explain how the criteria are being met.

Criteria (from checksheet)	**How course meets spirit (contextualize specific examples in next column)**	**Please provide detailed evidence of how course meets criteria (i.e., where in syllabus)**
1.a. Course involves the use of computer programming languages or software programs for quantitative analysis, modeling, simulation, animation, or statistics.	1. The attainment of basic computer programming constructs is one of 7 main learning outcomes of the course 2. Weeks 6-11 online lectures provide the students with fundamental computational knowledge. 3. Students' final project requires the use of a programming language.	1. Documentation: Syllabus, p1. Learning outcome #5 is "The student will identify and define basic procedural programming constructs used in MaxMSP and Processing." 2. Documentation: Course Topic Outline. Refer to lectures 6-11. Weeks 6 & 7, Experiential Media Systems, introduce students to systems theory and audio and visual feedback design in interactive systems. Week 8, Introduction to Programming, introduces common programming vocabulary terms such as syntax, variables, loops etc., as well as discussing types of programming languages and families, like graphical interface programs. Week 9, Introduction to MaxMSP and Processing, requires students to complete tutorials and readings about the graphical programming language MaxMSP. Week 10, Logistics of Programming in Media Arts, introduces students to the concept of open source software and briefly introduces Arduino, Pure Data, SuperCollider, ChucK, Quartz Composer, Scratch, Rhino, Grasshopper, and Firefly. Week 11, Physical Computing, demonstrates arduino hardware and software and its interfacing with MaxMSP and various sensors such as the Wii remote and photosensors. The lab classes

		paired with these lectures reinforce the concepts presented online. 3. Documentation: Final Project Handout. Unit 3 lab meetings are devoted to designing and implementing a final project. In order to complete a computational prototype of their design of an experiential media system, students must use a language of their choice to create a functional computational example of an interactive system they design as their final project. The final projects vary in nature; some are gaming systems, some are mobile apps, others are creative interventions for use in health or education.
1.b.ii Graphic/Artistic Design using computers	1. Students complete a hands-on, exploration based project with motion capture using frame differencing (or absolute difference). 2. Students are given an overview to Processing and provided the opportunity to work with the software for visual and interaction design. 3. Students are informed about open source software. 4. Students use a programming language to construct their final project, the design of an experiential media system.	1. Documentation: "Motion Capture Student Assignment.pdf." These slides accompany an in-class activity in which the students manipulate a MaxMSP patch and a Processing Sketch that both do frame differencing, or finding the motion between frames in a video as a form of motion detection. On the slides are the questions the students answer in class; for example, "What do you see when you change the frame rate? Why do you think this is? What do you see when you change the threshold? Why do you think this is? Can you come up with a definition for threshold just by interacting with it?" Which feedback modes feel more natural to you? Which do you prefer?" The latter question addresses artistic feedback design that is crucial in experiential media systems. An accompanying homework assignment is detailed on the last page and asks a series of questions about the activity, including, "In 2-5 sentences, describe the process of frame differencing and how it is used

		as motion capture." 2. Documentation: Assigned readings "Overview Of Processing" and "Getting Started With Processing." Processing is introduced in an online lecture called "Introduction to Programming" (see course topic outline), and demonsrated in depth at that week's accompanying lab meeting. Students are asked to complete basic processing or MaxMSP tutorials the following week. 3. Documentation: Course Topic Outline. Professor Loren Olson guest lectures on open source languages in the week called Logistics of Programming in Media Arts. Using open source is key to an artist when learning software rights and responsibilities especially borrowing from existing artist work. 4. Documentation: Course Topic Outline. Refer to lectures 6 & 7, which introduce students to a model for experiential media systems, including details of each component of the model. Feedback design focuses on visual and audio design with a focus on both aethestics and information display.
1.b.iii Music Design using computer software	1. Students gain familiarity with MaxMSP, a graphical programming language originally developed specifically for audio programming. 2. In Unit 3, Students use a programming language to construct their final project, the design of an experiential media system. 3. Students are introduced to and tested on fundamentals of audio knowledge in Unit 1.	1. Documentation: Assigned Readings from the Cycling '74 website titled, "What is Max?" and "Max is powerful software." See also, Course Topic Outline, and refer to week 9, Introduction to MaxMSP and Processing. 2a. Documentation: Final Project Student Handout. Goal 5 is "computation: the functional prototype demonstrates a small part of the designed system." On the same document, reference the project deliverables; one of

		them is a "computational prototype" to be developed in student-chosen programming language. Projects are required to have either visuals or audio as the feedback component of their system design. 2b. Documentation: Course Topic Outline. Refer to lectures 6 & 7, which introduce students to a model for experiential media systems, including details of each component of the model. Feedback design focuses on visual and audio design with a focus on both aethestics and information display. Music design is thus crucial to any audio based feedback in an experiential media system. 3. Documentation: Course Topic Outline, and lecture script "Script 2A-AudioRecordingTechniques." The lecture for week 2 is titled, "Sensing and Storytelling in Media Editing" and introduces the fundamentals of audio and video, including recording and capturing techniques for both mediums. For example, one part of the lecture reads, "How do microphones work? If sound hits a microphone, something similar to the process in the ear happens - changes in pressure vibrate a thin membrane called the diaphragm, which is attached to a coil, and the movement of the diaphragm converts creates an electrical signal in which the voltage and current are proportional to the original sound." Other topics from that script include noisefloor, amplitude, frequency, and types of microphones.

AME 111: Introduction to Digital Culture

Spring 2014

Time and Place: Online; Monday / Wednesday 1:30pm-2:45pm, Stauffer B125

Office Hours: TBA. By appointment; please contact via email to schedule a time.

Instructors: Kimberlee Swisher: kimberlee.swisher@asu.edu
Christian Ziegler: chris.ziegler@asu.edu
Nicole Williams (TA): nmwilliams@asu.edu

Course Objectives:

Students will learn about the fundamental structures and themes of Digital Culture. They will be introduced to examples of pathways for becoming active producers and consumers of Digital Culture through lecture, discussion, and hands-on activities. The course will cover key theoretical concepts of digital culture, fundamentals of media capture, composition and distribution as well as the basics of creating media systems in Processing and Max/MSP.

Learning Outcomes:

1. Students will identify and define the Dimensions of Digital Culture and place cultural artifacts along each of these continuums.
2. Students will capture, edit, and share digital audio and video online.
3. Students will define the School of Arts Media + Engineering's Experiential System Model, including sensing, modeling/analysis, and feedback towards an experiential goal.
4. Students will define the term "feature" (in the context of sensing/modeling) and, given an experiential goal, select appropriate features and analysis approaches.
5. The student will identify and define basic procedural programming constructs in MaxMSP and Processing.
6. Students will write about the current landscape of Digital Culture practice, demonstrating understanding of key terms and current and future trends.
7. Students will, as part of a team of peers, propose an experiential goal and contribute to the design of an experiential media system that implements sensing, modeling, and feedback (using instructor-provided Max/MSP patches) in support of this goal.

Incoming Proficiencies

This course has no required incoming proficiencies. In future Digital Culture classes, the following applies:

Enrolling in this course assumes that you have attained the necessary Digital Culture proficiencies required for this course. It is your responsibility to create and follow your Digital Culture course planner path. Your professor may require you to print out your path as proof that you have the knowledge necessary to be successful in their course. Extra class time will not be spent making up deficiencies in knowledge due to a disregard of the course planner. If you know that you do not meet the proficiencies, please drop the course now. If you are unsure of what proficiencies you need for the class, please go here to create a path: https://digitalculture.asu.edu/map.

Outgoing Proficiencies

Passing this course with the minimum grade required by the department grants the following Digital Culture outgoing proficiencies:

Collaborative Principles	100
Cultural Practice and Studies	100
Form and Composition	100
Modeling and Inference	100
Social Mechanisms and Understanding	100

Class Expectations:

Assignment submissions:

All assignments must be submitted in the correct *format* and *place* per the assignment instructions, or they will not count for credit. Emailed assignments do not count unless specifically designated.

Conduct:

Use of cellphones or other electronic devices is not permitted during class. Laptop use is permitted with instructor permission. Threatening, violent or disruptive conduct will not be tolerated under any circumstances. If any student commits violent, threatening or disruptive acts, he or she will be asked to leave and reported to the appropriate authorities, including ASU Campus Police and the Office of Student Rights and Responsibilities. This applies to any conduct, including both virtual and in-person communication with instructors, TAs, guests, and other colleagues.

Group work:

This course involves a substantial amount of collaborative work. We expect that all members of the group will contribute equally to their projects. We reserve the right to assign individual grades based on individual contributions to group projects.

Written assignments

Proper grammar, spelling, sentence structure, professionalism, style and other writing skills are required in all written documents completed for this course, including but not limited to: discussion posts, extra credit, projects, papers, and emails to instructors. Graded assignments may be marked off if they do not meet one or more of these requirements.

Email Communication

Who to contact

All technical questions about the course should be addressed to Herberger Online. Emails sent to the instructors about technical issues may not be answered. Emails regarding attendance should be sent to the TA (nmwilliams@asu.edu). General course questions may be addressed to any or all instructors.

Response time

Please allow 24-48 hours for an email reply from instructors or Herberger Online. Emails received by Herberger Online or the instructor after 5:00m on Friday may not be answered until after 8:00am the following Monday.

Content

Any emails requesting exceptions to the class policies as outlined on this document will not be answered. Any emails that ask questions that have already been answered on the syllabus will not be answered. We also reserve the right to not answer any rude, argumentative, threatening, abusive or repetitive emails, and to forward them to the proper authorities as necessary. Instructors reserve the right to request in-person communication instead of email.

Student responsibilities

As an ASU student, you are required to use your ASUrite (i.e., student@asu.edu) for course communications. If our reply to you bounces back as "Undeliverable: Returned mail" and/or with "the message could not be delivered because the recipient's mailbox is full," we will not make a second attempt. This includes any and all course communications, such as make-up verifications and make-up exams. It is your responsibility to make sure you are able to receive ASU emails by clearing your mailbox and allowing ASU and MUS 354 emails to be received unblocked and not marked as spam.

CritViz

Overview

We will be using an online service called CritViz for two assignments in this course. CritViz was created by David Tinapple and Loren Olson, two professors here at the School of Arts, Media and Engineering. These two assignments happen in unit 2 and there will be an announcement at that time indicating the deadline for signing-up for CritViz. Please wait for the announcement to sign-up.

How to sign-up when the time comes

Go to critviz.com and choose New User. Enter your asurite ID *only*. This is the name you use to login to my.asu.edu. Do NOT include the @asu.edu after the name. CritViz will send an email to your ASU account with further instructions. If you do not receive an email, you have entered your asurite ID incorrectly.

Getting Help

Please do not email Herberger Online for help with CritViz, as they do not own it and cannot help you. You may email the instructors with questions, but please note that if the questions are in regards to signing up, we will point you directly back to this page of the syllabus.

Tests and Quizzes

Unit 1 Quiz

This short exam will cover material from lectures and readings covered prior to the exam.

Midterm Exam

The midterm will cover material from the lectures and readings, as well as theoretical concepts illustrated through the projects and in-class / homework assignments. It will be cumulative, meaning that it DOES include material that was covered on the first exam.

Pop Quizzes

Pop quizzes may occur at anytime, and cannot be made up in the case of an unexcused absence.

Final Exam

There is no final exam in this course. However, we *will* hold class on the university mandated final exam day, Monday May 5th 12:10p - 2p (available at https://students.asu.edu/final-exam-schedule).

Projects

Project I: Capturing and Presenting Human Activity Through Digital Media

This group project consists of recording, editing and producing audio and video that captures a complex human activity. Students will work in teams to complete the media artifact and a

300-500 word reflection paper. The project will be evaluated based on: 1) the quality of the reflection paper; 2) technical proficiency of recording and editing; 3) form and composition of the final media artifact; and 4) effectiveness of capturing and presenting the complex human activity.

Final Project: Real-Time Interactivity in Hybrid (Physical-Digital) Systems

For the final project, students work in teams to create a real-time interactive media system. Each student will be evaluated based on his or her contribution to: 1) a group reflection paper; 2) the functioning interactive system's code; 3) a 3-5 minute video demonstration of the system; 4) a group presentation in class; and 5) an individual reflection paper.

Grade Weights:

40% Projects (15% Project I, 25% Final Project)
30% Pop Quizzes, In-Class Assignments, Homework, Discussion Board Postings
20% Exams (10% unit 1, 15% Midterm)
10% Class Participation + Attendance

Grading Scale:

A+: 97-100%
A: 93-96.9%
A-: 90-92.9%
B+: 87-89.9%
B: 83-87.9%
B-: 80-82.9%
C+: 77-79.9%
C: 70-76.9%
D: 60-69.9%
E: 0-59.9%

A curve may be applied to raw scores at the instructor's discretion.

Grading Policy

It is the responsibility of the student to check their grades regularly and make any concerns known to the instructors. The instructors may periodically impose deadlines for students to inform instructors of any grade concerns, and any graded material prior to this deadline will not be changed after the deadline.

Attendance Policy

Students are expected to attend all class sessions. Students must sign in on the attendance sheet at the beginning of each class. Up to **two** unexcused absences are permitted per semester. After **two** unexcused absences, 25% of the participation score will be deducted per additional unexcused absence until participation grade reaches 0%. In extraordinary circumstances where more than six unexcused absences occur in combination with poor participation, the instructors reserve the right to deduct 2.5% per unexcused absence from the overall semester grade, over and above issuing a 0% for participation.

Absences may be excused for religious observance and for university-sanctioned events. Please email the instructor 48 hours before taking an excused absence under either of these policies. For illness or medical emergency, please contact the instructor via email as soon as possible and be prepared to provide a note from your doctor. Students are required to submit all assignments at the beginning of the next class session after an excused absence; after this, normal late penalties (described below) apply. Reasonable accommodations will be made in the event of medical emergency or extended excused absence.

Late Assignment Policy

In-class assignments are an important part of the course, and rely on interaction that can't be reproduced after the fact. For this reason, they cannot be made up in the event of unexcused absences. In the case of an excused absence, the instructor may provide an alternative assignment, or allow the late completion of the in-class assignment.

Projects and assignments must be turned in promptly. 25% of the graded score will be deducted for each 24-hour period past the deadline. Per our assignment submission policy (see Class Expectations), assignments that are submitted in the incorrect format or place will not count and will thus be considered late until resubmitted correctly.

Exams

Overview

There will be two exams, all taken online. The first is a shorter Unit 1 exam; the second is the midterm and is cumulative. The computer, from a large question bank for each and every exam, randomly chooses all exam questions. All exam questions are drawn directly from the lecture materials and assigned readings.

Academic Integrity

You should NOT use any written, online or other notes during the exam itself, including opening any additional web pages from the course or other locations, or collaborating with other students. Tests are not to be taken as a group effort. These activities are in direct violation of university and course policies. Students found cheating will be sanctioned in accordance with the student code of conduct and the rules against academic dishonesty upheld and enforced by Arizona State University. Please visit the following sites for detailed information:
http://www.asu.edu/studentaffairs/studentlife/judicial/academic_integrity.htm

Do not give your ID or password to anyone else, and do not allow anyone else to access or take exams or assignments in your name. No collaboration or use of outside sources of any kind is allowed on any tests, assignments, etc. This includes viewing downloaded course website information while taking the exam, having any Internet sites open other than the exam link, listening to audio excerpts on another student's exam, discussing the exam with another student before or while taking it, using notes during an exam, and other such behavior.

Any violation of this policy will result in sanctions and may result in further disciplinary action. All violations of the Code detected during the term, either electronically or by other means, will be sanctioned, even if the exam has already been graded and points assigned. Please note that our computer logs record all student interactions with the course website, and Herberger Online randomly and specifically screens these logs for evidence of collaboration and cheating. If such activity is observed either physically or through computer records, all students involved will be called to account for their actions. Sanctions will be in accordance with the student code of conduct and the rules against academic dishonesty as upheld and enforced by Arizona State University. There will be no exceptions to these procedures or leniency regarding these policies.

NOTE: NO course content will be available to students while they are taking an exam. Once you enter the exam, you will NOT be able to access course materials on the website. Please refrain from opening any additional tabs/windows on your browser while taking an exam, as this could cause your submission to be unsuccessful. You will need to study and access the specific section lectures before starting an exam. There will be no access to these materials for any students while taking an exam, without exception. We appreciate your understanding and cooperation and apologize for any inconvenience.

Testing Procedures

It is your responsibility to locate reliable Internet access during the exam. Make-ups will not be available for students who have computer difficulties at their end. Starting the exam during the last day is not recommended, as you may go encounter technical difficulties.

You will be issued a unique receipt number for every successfully submitted exam. Please retain that number through the semester as proof of your completed submission. If you do not receive a receipt number for an exam or assignment, that submission has NOT been successful, and you should contact Herberger Online immediately at http://herbergeronline.asu.edu/help for assistance. If you have no receipt number, you have no grade. It is your responsibility to print or write out each receipt number you receive for each exam/topic assignment you submit. This is your official receipt for work completed, so it is important to print and keep these receipts. Please email Herberger Online if your score is not posted on the course Grades page within 48 hours after you complete the exam. Inquiries about missing grades sent to faculty, staff, or to Herberger Online will be ignored if they do not have the receipt number(s) for the exam/assignment in question.

Do not use iPhones, blackberries, tablets, or any other palm devices to take an exam, and wireless connections are not recommended. Tests will not be accepted in printed form or via email or email attachment.

Tips to avoid complications while taking an exam:

- It is best to log in to the exam and complete it without leaving the exam page at all. Opening other tabs or windows, especially pages from the class website, can lead to problems when submitting the exam.
- Do not reload/refresh the exam page during the exam.
- Avoid logging into or out of any other ASU page (myASU, etc.) during the exam, as your login session is tied to your ASU account, and logging out of any other ASU page during the exam will log you out of the exam as well. In other words, treat this like an exam in a normal classroom - log in and start the exam, and don't do anything else until you’ve finished.
- Most of the technical problems Herberger Online encounters regarding exams stem from students opening multiple tabs and logging in and out of the class during an exam.

Resets and Make-Up Exams

If you experience computer or technical problems during an exam, DO NOT JUST GUESS, OPEN ADDITIONAL WINDOWS, OR SUBMIT IT, as that grade must stand as posted! Instead, exit your browser, fix the problem (move to a different computer or contact technical support by using the 'Submit a Help Request' button on the course Help page, or by going directly to https://herbergeronline.asu.edu/help), and then return to the exams page on the course website to reset your exam. The ASU Help Desk cannot assist you with this course, so please email Herberger Online for help. The system will allow you to reset your own exam if you have not already submitted or reset that exam. We recommend, in these cases, that you restart your browser before you try to reset the exam. Typically, you can reset exams by yourself simply by logging back into the exam as you normally would. When you go back to the exams page, the system will know that you need a reset, and you just have to follow the links provided.

Resetting any exam means the computer will be creating a new exam for you. Any answers you provided before are already lost, and you will be starting over again. All exams draw random questions from a question bank, so you will not get the same exact questions in the same order again.

If you experience personal, medical or other unforeseen problems during the exam period, DO NOT TAKE OR SUBMIT AN EXAM, as that grade must stand as posted! Instead, email the instructors immediately to discuss both your situation and the process necessary for a make-up exam. Exams that have been submitted CANNOT be reset, and those grades must stand as posted. There are no re-dos or retakes on submitted exams under any circumstances!
Resets are only available for students experiencing technical difficulties while testing online. Only one online reset per exam is allowed, and only two resets or make-ups per student per semester on two different exams are allowed. If you are not able to successfully submit your reset exam due to technical reasons verifiable by Herberger Online, you may have the option of an essay style make-up exam. This option will also be available to students presenting verifiable documentation regarding illness, or other personal issues. The instructor reserves the right to substitute an online make-up exam with an essay-style make-up exam at any time without question.

If you need a make-up exam, please email the instructor and be prepared to electronically present an official document on printed or company letterhead (with a logo, not your personal letterhead). This official document must contain a contact name and phone number and must be from either 1) a doctor or hospital if you of someone from your immediate family are ill, 2) a funeral announcement from a church or funeral home if someone dies, 3) an airline ticket containing your unique information if you are required to travel for personal or business reasons, 4) an employer letter if you are required to miss exams for business or work-related duties, 5) a letter from your coach or academic advisor detailing your sport, days absent, and reasons why you were unable to obtain internet access during the exam period, or 6) any other verifiable sources who can detail specifics as to why you were unable to complete and submit your exam during the five weeks it was available and are now requesting an essay make-up exam.

You will need to contact the instructors as soon as possible for make-up consideration. Please also indicate which class you are in at that time. No make-ups will be given until our staff has completed phone verification of your official document. After receipt of this document and our verification of its contents, we will contact you again via email with specific make-up instructions. All submitted documentation will be checked for authenticity; those who submit false documentation will be sanctioned according to the Student Code of Conduct. Please do not submit false documentation.

The score for this paper will act as the score for the failed reset, or as your make-up exam. All words and ideas included in your essay exam must be your own or sanctions will be levied per the Student Code of Conduct. All essay-style make-ups must be secured by instructor permission only, and are only allowed under specific, documented circumstances. There will be no exceptions to this procedure.

Academic Integrity

All necessary sanctions will be issued to all parties involved with plagiarizing any and all course work. Plagiarism and any other form of academic dishonesty that is in violation with the Student Code of Conduct will not be tolerated. For more information, please see the ASU Student Academic Integrity Policy: http://www.asu.edu/studentaffairs/studentlife/judicial/academic_integrity.htm. *Please note that work submitted for assessment in one class may not be submitted in full or in part for assessment in a second class.*

Accommodations for Students with Disabilities

To request academic accommodations due to a disability, please contact the ASU Disability Resource Center (http://www.asu.edu/studentaffairs/ed/drc/#); Phone: (480) 965-1234; TDD: (480) 965-9000). This is a very important step as accommodations may be difficult to make retroactively. If you have a letter from their office indicating that you have a disability which requires academic accommodations, in order to assure that you receive your accommodations in a timely manner, please present this documentation no later than the two weeks into the semester so that your needs can be addressed effectively.

Incompletes

Incompletes are awarded under only very special circumstances and are only considered for students in good standing. Each incomplete is considered on a case-by-case basis to determine the extent to which they are warranted. The "Request for Incomplete" form must be signed and submitted by the instructor to the School of Arts Media + Engineering office for final approval by the School Director. The incomplete form can be downloaded from the following website: http://students.asu.edu/forms/incomplete-grade-request.

Important Dates

Spring Break: March 9th -16th, 2014
University Course Withdrawal Deadline: April 6th, 2014
Last Day of Classes: Friday, May 2nd, 2014
Final Exam: Monday, May 5th, 12:10p-2pm

No exceptions will be made to any of the policies as outlined in this document. Any changes to course information made by the instructor or staff will be posted on this syllabus and/or the course homepage, and both sites are subject to change at anytime, without notice. Please check your ASU email account and both the syllabus and class homepage frequently for updated information and announcements.

Please consider this document your contract with AME 111. By enrolling and participating you agree to be obligated to follow the policies and procedures of this course and abide by all of the terms as outlined without question or complaint. You agree to regularly reread this document as well as the class homepage for new information. We appreciate your patience with the inadvertently brusque and direct nature and tone of this syllabus deemed necessary to be effective with so many students in this course. We also greatly appreciate your patience and assistance throughout the course, as this is the first run of this online course and certain procedures may still need to be unkinked.

Schedule of Deadlines

All deadlines are subject to change

Friday **January 17th, 11:59pm**

Watch Lecture 0, Course Introduction

Friday **January 17th, 11:59pm**

Submit Discussion 0: Lecture 0

Mon/Wed **January 20th/22nd, class time**

Watch Lecture 1, Dimensions of Digital Culture

Read During, Simon. (1993). *Cultural Studies Reader*. p.1-3 and 22-25.

Read McLuhan, Marshall (1966). *Medium is the Message*. Speech, transcribed as a chapter in *Understanding Me*.

Friday **January 24th, 11:59pm**

Submit Discussion 1: Lecture 1

Mon/Wed **January 27th/29th, class time**

Watch lecture 2, Sensing and Storytelling with Media Editing

Friday **January 31st, 11:59pm**

Must be in a project I group!

Mon/Wed **February 3rd/5th, class time**

Watch lecture 3, Networks

Friday **February 7th, 11:59pm**

Submit Discussion 2: Lecture 3

Submit pressure project documentation

Mon/Wed **February 10th/12th, class time**

Watch lecture 4, Creative Practice in Digital Culture

Optional reading: Kirn, P. (2013). When New Media is no Longer New, Disciplines Blur. http://createdigitalmotion.com/2013/07/when-new-media-is-no-longer-new-disciplines-blur-editorial/

Friday **February 14th, 11:59pm**

Project I DUE!

Mon/Wed **February 17th/19th, class time**

Watch lecture 5, Chris Anderson's The Long Tail

Read: Anderson, Chris. (2005). The Long Tail (Article in Wired)
Gladwell, Malcolm. (2009). Review of Free by Chris Anderson.

(UNIT 1 EXAM opens for business Monday Feb 17th at 6am)

Friday **February 21st, 11:59pm**

Submit Discussion 3: Lecture 5

UNIT 1 EXAM DUE

Mon/Wed **February 24th/26th, class time**

Watch lecture 6, Experiential Media Systems Part 1

Read Davis, Marc. (2003). *Theoretical Foundations of Experiential Media Systems*.

Friday **February 28th, 11:59pm**

Submit Frame Differencing Assignment (CritViz)

Mon/Wed **March 3rd/5th, class time**

Watch lecture 7 Experiential Media Systems Part 2

Read: Dourish, Paul. (2001). *Where The Action Is*. p. 1-3 & 17-20, optional: the rest of chapter

Friday **March 7th, 11:59pm**

Submit Frame Differencing *Critique* (CritViz)

Submit Discussion 4: Lecture 7

SPRING BREAK : March 9TH - 16TH, 2014

Friday, **March 21st, 2014 (Deadline for watching this lecture pushed back from normal "class time" to Friday evening in order to avoid the necessity of Spring Break homework)**

Watch lecture 8 Introduction to Programming

Mon/Wed **March 24th/26th, class time**

Complete activities for lecture 9 Introduction to MaxMSP and Processing

Friday **March 28th, 11:59pm**

Submit Discussion 5: Lecture 9

Complete 5 max msp tutorials or 5 processing tutorials on your own.

Mon/Wed **March 31st/April 2nd, class time**

Watch lecture 10 Logistics of Programming in Media Arts

(MIDTERM EXAM opens for business Monday March 31st at 6am)

Friday **April 4th, 11:59pm**

MIDTERM EXAM DUE!

Mon/Wed **April 7th/9th class time**

Watch lecture 11 Physical Computing

Friday **April 11th, 11:59pm**

Submit Discussion 6: Lecture 11

Mon/Wed **April 14th/16th class time**

Watch lecture 12 The Micro-Macro Link and Emergence

Read: Sawyer, R. K. (2005). *Social Emergence: Societies As Complex Systems*. p. 5-9.

Friday **April 18th, 11:59pm**

Submit Critiques of Final Project Proposal (CritViz)

Mon/Wed **April 21st/23rd class time**

Watch lecture 13 Ubiquitous Computing and the Programmable Environment

Read: Herman Miller Group. (2008). *Always Building. p. 1-50.*

Friday **April 25th, 11:59pm**

Submit Discussion 7: Lecture 13

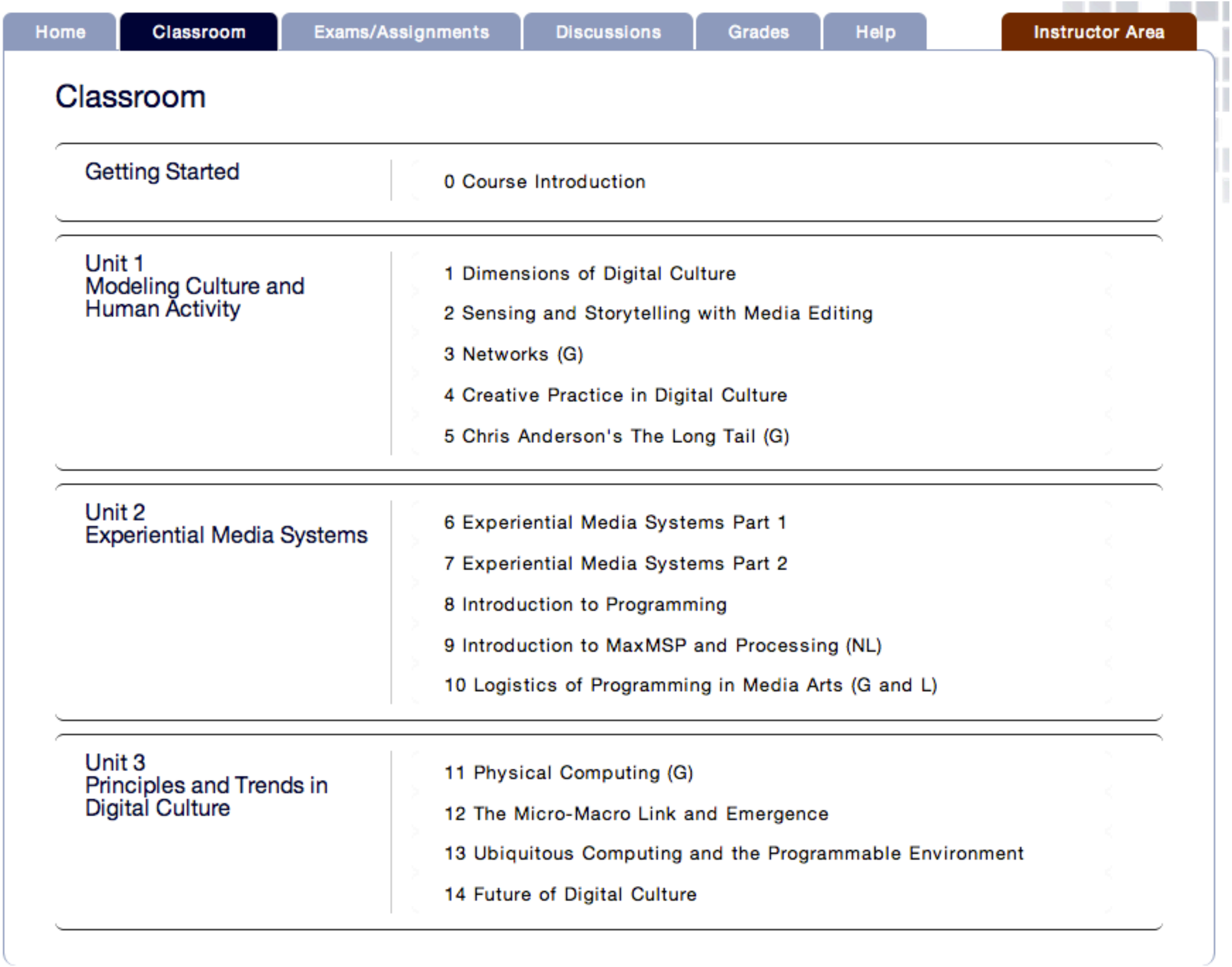
Mon/Wed **April 21st/23rd class time**

Complete activities for lecture 14, Future of Digital Culture

Friday **May 2nd, 11:59pm**

FINAL PROJECT DUE

This is a screenshot from our online course classroom, and depicts all of the lecture topics week by week. Each lecture title is a link to that week's course content.



Classroom

Getting Started

0 Course Introduction

Unit 1
Modeling Culture and Human Activity

1 Dimensions of Digital Culture
2 Sensing and Storytelling with Media Editing
3 Networks (G)
4 Creative Practice in Digital Culture
5 Chris Anderson's The Long Tail (G)

Unit 2
Experiential Media Systems

6 Experiential Media Systems Part 1
7 Experiential Media Systems Part 2
8 Introduction to Programming
9 Introduction to MaxMSP and Processing (NL)
10 Logistics of Programming in Media Arts (G and L)

Unit 3
Principles and Trends in Digital Culture

11 Physical Computing (G)
12 The Micro-Macro Link and Emergence
13 Ubiquitous Computing and the Programmable Environment
14 Future of Digital Culture

AME 111: Introduction to Digital Culture
Course Readings
There is no required text for this class. Course readings come from various printed and electronic sources and will be provided to you. For written articles, PDFs will be provided.

Print sources:
During, Simon. (1993). *Cultural Studies Reader*. p.1-3 and 22-25.
McLuhan, Marshall (1966). *Medium is the Message*. Speech, transcribed as a chapter in *Understanding Me*.
Anderson, Chris. (2005). The Long Tail (Article in Wired)
Gladwell, Malcolm. (2009). Review of Free by Chris Anderson.
Dourish, Paul. (2001). *Where The Action Is*. p. 1-3 & 17-20, optional: the rest of chapter
Davis, Marc. (2003). *Theoretical Foundations of Experiential Media Systems*.
Sawyer, R. K. (2005). *Social Emergence: Societies As Complex Systems*. p. 5-9. *(optional)*
Rikakis, Thanassis. (2010). Post-Disciplinary Liberal Education. p.1-2
Jaimes, Alejandro.(2006). Human-Centered Multimedia
Kirn, P. (2013). When New Media is no Longer New, Disciplines Blur.
http://createdigitalmotion.com/2013/07/when-new-media-is-no-longer-new-disciplines-blur-editorial/
Scott Page. (2008) *The Difference*. p.180-185

Browsing resources:
http://processing.org/overview/
http://processing.org/tutorials/gettingstarted/
http://cycling74.com/whatismax/
http://cycling74.com/products/max/

AME 111, Fall 2013
Final Project

Task

In a self-selected team of 3-5 classmates, propose, design, prototype and document *a real-time, experiential, interactive system* that is either an 1) **OBJECT** 2) **SPACE** or 3) **PROCESS** [or some combination of all three that you can describe].

Available Tools

Anything in the DC equipment lab, including but not limited to:

- DC computers, laptop cameras, microphones, video cameras, Kinect, Wii Remotes.
- MaxMSP, Processing, Osculator, Audacity, iMovie, Photoshop, Madmapper, Ableton Live

Deliverables:

+ Project Proposal *(Due November 12th at 11:59pm)* | Homework grade

There is a specific format for this, given as a group assignment page on Blackboard. This counts as a homework assignment and not as part of the overall project grade.

This project is worth 50 points total (25% of your grade). Included in the project grade are the following:

+ Video Demo *(Due December 9th at 11:59pm)* | 20 points

Make a 1-5 minute video that explains how the system works and demonstrates the system in use. This should show the design, functionality, and compelling nature of the proposed system. As the basis of the project is a proposal and not a fully functioning system; you can have some content be a facade.

+ Computational prototype of one aspect of your project *(Due December 9th at 11:59pm)*| 5 points

We will provide starter patches in MaxMSP for less experienced programmers. Choose one interactive aspect of your project to prototype computationally in your choice of programming languages.

+ Group Written Documentation *(Due on day of final)* | 10 points

5 pages **maximum** (12pt font, double-spaced). Detail your system. Describe its purpose and how it works. Explain the sensing, analysis, and feedback, and discuss the positive and negative aspects of each one as it relates to your specific system. Discuss the social interactions among the system's users and how the system influences them. Also address the interaction between the users and the system itself. Talk about the aspects of your system that make it compelling and engaging. Discuss future iterations of the project (how it could be expended, what "real" situations it would be applied in, and if there are other uses that came up in development or discussion that you did not anticipate). Describe how you incorporated feedback from colleagues into the iterative development of your system.

+ Individual Reflection *(Due on day of final)* | 10 points

2 pages **maximum** (12pt font, double-spaced). Give us a very short elevator pitch of your system, and then explain your individual contribution to the group project. Reflect on your group's dynamics and the collaborative development process. Include anything you feel that the instructors should know that was personally relevant to your whole experience.

+ Group Demo/Presentation *(Due on day of final)* | 5 points

On the day of the final, each group will participate in a poster session presentation to classmates, instructors, and guest faculty. We hope to see a working demo, your documentation video, and any other artifacts you deem necessary to pitch your project. Each group member should be knowledgeable enough to answer questions, and should partake in the presentation.

Goals

1. **Experience**: The system is compelling, purposeful and has a clear experiential goal. The system incorporates real-time sensing and analysis to produce interactivity and media feedback.
2. **Interaction**: the system is engaging and promotes interactivity; between person-person or person-computer.
3. **Collaboration**: Every group member participates in creating the system, and has an explicit and critical role to play in its development.
4. **Design**: The system is thoughtfully and iteratively designed, incorporating knowledge from multiple trials. The video documentation is professional and effectively shows your system.
5. **Computation**: The functional prototype demonstrates a small part of the designed system.

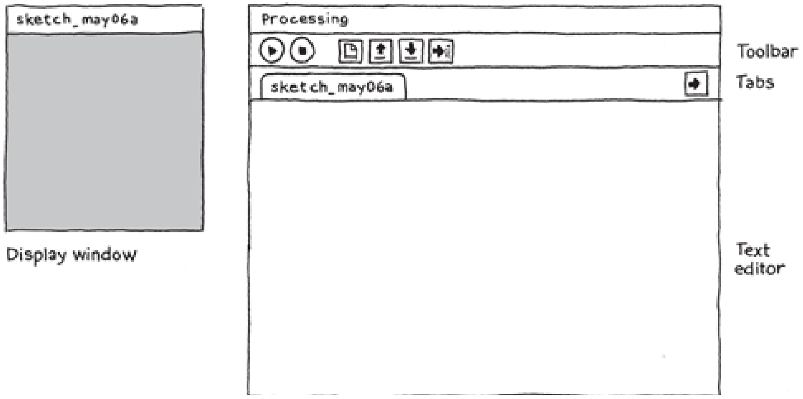
Getting Started. Welcome to Processing!

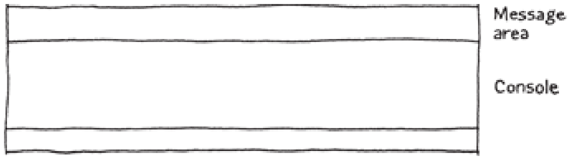
This tutorial is for Processing 2+. If you see any errors or have comments, please let us know. This tutorial was adapted from the book, Getting Started with Processing, by Casey Reas and Ben Fry, O'Reilly / Make 2010.

Start by visiting http://processing.org/download and selecting the Mac, Windows, or Linux version, depending on what machine you have. Installation on each machine is straightforward:

- On Windows, you'll have a .zip file. Double-click it, and drag the folder inside to a location on your hard disk. It could be Program Files or simply the desktop, but the important thing is for the processing folder to be pulled out of that .zip file. Then double-click processing.exe to start.
- The Mac OS X version is also a .zip file. Double-click it and drag the Processing icon to the Applications folder. If you're using someone else's machine and can't modify the Applications folder, just drag the application to the desktop. Then double-click the Processing icon to start.
- The Linux version is a .tar.gz file, which should be familiar to most Linux users. Download the file to your home directory, then open a terminal window, and type:
  ```
  tar xvfz processing-xxxx.tgz
  ```
  (Replace xxxx with the rest of the file's name, which is the version number.) This will create a folder named processing-2.0 or something similar. Then change to that directory:
  ```
  cd processing-xxxx
  ```
  and run it:
  ```
  ./processing
  ```

With any luck, the main Processing window will now be visible. Everyone's setup is different, so if the program didn't start, or you're otherwise stuck, visit the troubleshooting page for possible solutions.





The Processing Development Environment.

Your First Program

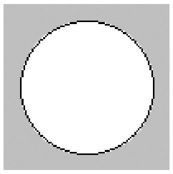
You're now running the Processing Development Environment (or PDE). There's not much to it; the large area is the Text Editor, and there's a row of buttons across the top; this is the toolbar. Below the editor is the Message Area, and below that is the Console. The Message Area is used for one line messages, and the Console is used for more technical details.

In the editor, type the following:

```
ellipse(50, 50, 80, 80);
```

This line of code means "draw an ellipse, with the center 50 pixels over from the left and 50 pixels down from the top, with a width and height of 80 pixels." Click the Run button, which looks like this:

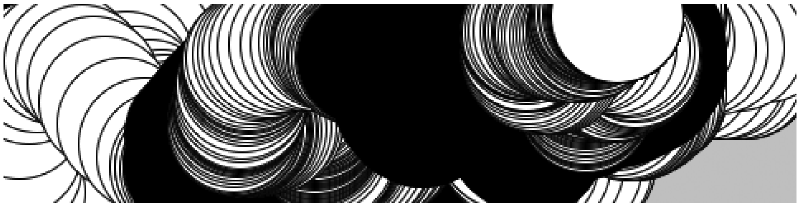
If you've typed everything correctly, you'll see this appear in the Display Window:



If you didn't type it correctly, the Message Area will turn red and complain about an error. If this happens, make sure that you've copied the example code exactly: the numbers should be contained within parentheses and have commas between each of them, and the line should end with a semicolon.

One of the most difficult things about getting started with programming is that you have to be very specific about the syntax. The Processing software isn't always smart enough to know what you mean, and can be quite fussy about the placement of punctuation. You'll get used to it with a little practice.

Next, we'll skip ahead to a sketch that's a little more exciting. Delete the text from the last example, and try this:



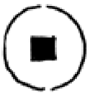
```
void setup() {
  size(480, 120);
}

void draw() {
  if (mousePressed) {
    fill(0);
  } else {
    fill(255);
  }
  ellipse(mouseX, mouseY, 80, 80);
}
```

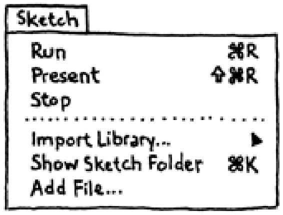
This program creates a window that is 480 pixels wide and 120 pixels high, and then starts drawing white circles at the position of the mouse. When a mouse button is pressed, the circle color changes to black. We'll explain more about the elements of this program in detail later. For now, run the code, move the mouse, and click to experience it.

Show

So far we've covered only the Run button, though you've probably guessed what the Stop button next to it does:



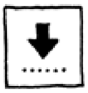
If you don't want to use the buttons, you can always use the Sketch menu, which reveals the shortcut Ctrl-R (or Cmd-R on the Mac) for Run. Below Run in the Sketch menu is Present, which clears the rest of the screen to present your sketch all by itself:



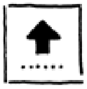
You can also use Present from the toolbar by holding down the Shift key as you click the Run button.

Save

The next command that's important is Save. It's the downward arrow on the toolbar:

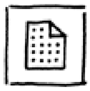


You can also find it under the File menu. By default, your programs are saved to the "sketchbook," which is a folder that collects your programs for easy access. Clicking the Open button on the toolbar (the arrow pointing up) will bring up a list of all the sketches in your sketchbook, as well as a list of examples that are installed with the Processing software:



It's always a good idea to save your sketches often. As you try different things, keep saving with different names, so that you can always go back to an earlier version. This is especially helpful if — no, when — something breaks. You can also see where the sketch is located on the disk with Show Sketch Folder under the Sketch menu.

You can also create a new sketch by pressing the New button on the toolbar:



Share

Share

Another theme of Processing is sharing your work. The Export Application button on the toolbar:

will bundle your code into an application for your choice of Mac, Windows, or Linux depending on which platform you are using. This is an easy way to make self-contained, double-clickable versions of your projects.

You can also find Export Application under the File menu.

In addition to exporting your code as applications, you can switch to a different *mode* within Processing to export to other platforms. For example, download and change to JavaScript Modeto export HTML5 Canvas and WebGL. Change to Android mode to export application for Android phones and tablets. These modes both need to be added before they can be used. Select "Add Mode..." from the menu that says "Java" in the upper-right corner of the Processing Development Environment.

Examples and Reference

Learning how to program with Processing involves exploring lots of code: running, altering, breaking, and enhancing it until you have reshaped it into something new. With this in mind, the Processing software download includes dozens of examples that demonstrate different features of the software. To open an example, select Examples from the File menu or click the Open icon in the PDE. The examples are grouped into categories based on their function, such as Form, Motion, and Image. Find an interesting topic in the list and try an example.

If you see a part of the program you're unfamiliar with that is colored orange (this means it's a part of Processing), select its name, and then click on "Find in Reference" from the Help menu. You can also right-click the text (or Ctrl-click on a Mac) and choose Find in Reference from the menu that appears. This will open the reference for the selected code element in your web browser. The reference is also available online.

The Processing Reference explains every code element with a description and examples. The reference programs are much shorter (usually four or five lines) and easier to follow than the longer code found in the Examples folder. We recommend keeping the reference open while you're reading this book and while you're programming. It can be navigated by topic or alphabetically; sometimes it's fastest to do a text search within your browser window.

The reference was written with the beginner in mind; we hope that we've made it clear and understandable. We're grateful to the many people who've spotted errors over the years and reported them. If you think you can improve a reference entry or you find a mistake, please let us know by clicking on the link at the top of each reference page.

This tutorial is for Processing 2+. If you see any errors or have comments, please let us know. This tutorial was adapted from the book, Getting Started with Processing, by Casey Reas and Ben Fry, O'Reilly / Make 2010.

Processing was initiated by Ben Fry and Casey Reas. It is developed by a small team of volunteers.

Frame Differencing in MaxMSP and Processing

October 9th & 10th, 2013

Intro to today’s activity

SENSING

Action:

Open the MaxMSP patch. Make sure it is locked (button in the lower left corner). Press “open” to start the camera.

Discuss:

What is the sensor in this system?

How are you controlling it in the max patch? (i.e. what do you have to do in the patch to control the sensor?

How do you know when it is working?

ANALYSIS

Action:

Move and play around in front of the camera. Observe the color video stream and the black and white video stream. Change the frame rate and threshold numbers and observe again.

Discuss:

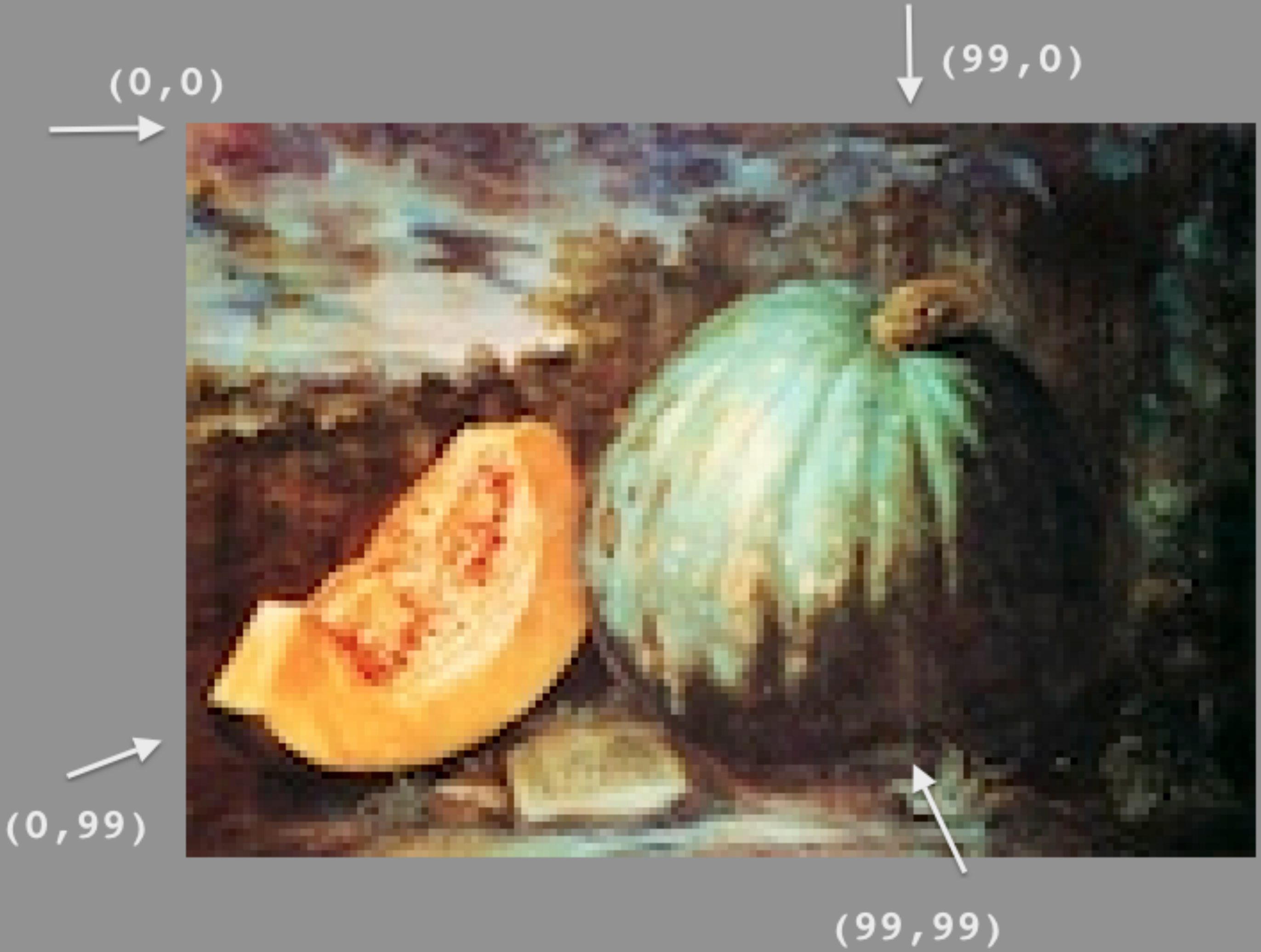
Which one is the sensing viewer? Which is the analysis viewer? What do you see in the analysis viewer?

Which button(s) affect the analysis?

What do you see when you change the frame rate? Why do you think this is?

What do you see when you change the threshold? Why do you think this is? Can you come up with a definition for threshold just by interacting with it?

REVIEW OF COLOR



(0,0)

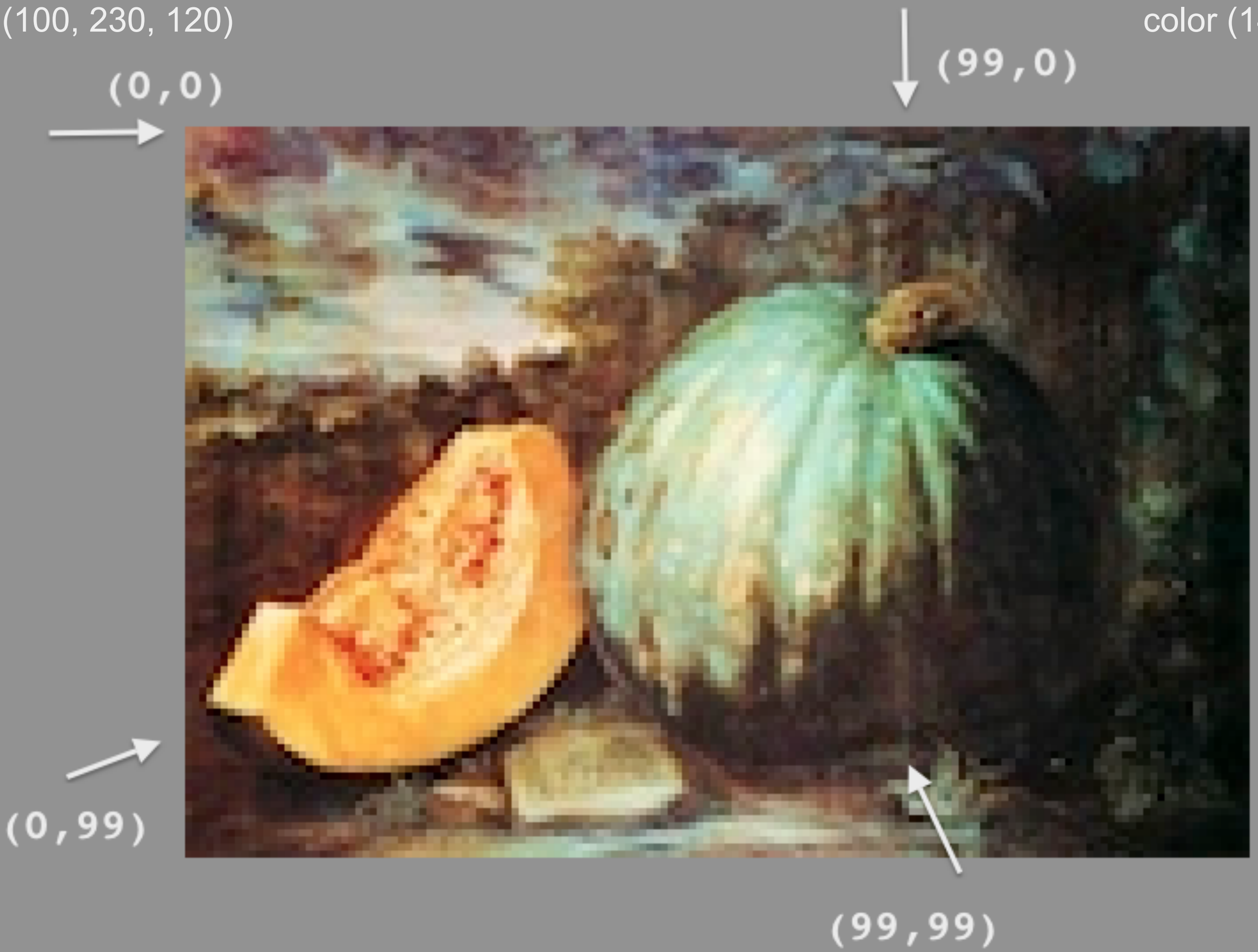
(99,0)

(0,99)

(99,99)

pixel (0, 0) has
color (100, 230, 120)

pixel (0, 0) has
color (14, 15, 45)



Zucca nata in Pisa
nel Giardino di S.A.R.
detto di S. Francesco
l'Anno 1711 Pesava lb.160.

now, all pixels can be defined by just one number, because it is grayscale.

ANALYSIS

Action:

Move and play around in front of the camera. Observe the color video stream and the black and white video stream. Change the frame rate and threshold numbers and observe again.

Discuss:

Which one is the sensing viewer? Which is the analysis viewer? What do you see in the analysis viewer?

Which button(s) affect the analysis?

What do you see when you change the frame rate? Why do you think this is?

What do you see when you change the threshold? Why do you think this is? Can you come up with a definition for threshold just by interacting with it?

So, What is Frame Differencing?

So, What is Frame Differencing?

Frame differencing is a method of detecting motion that compares the grayscale pixel colors (numbers) between each adjacent frame, subtracting them and thus determining the “difference” between each frame.

FEEDBACK

Action:

Change the feedback modes (fade vs. on/off and small vs. large lines) in the Max patch and interact with the system again.

Discuss:

Which feedback modes feel more natural to you? Which do you prefer?

How would you (and/or your team) change the feedback if given a chance to redesign the system?

How does changing the threshold and/or frame rate affect the legibility of the feedback?

PUTTING IT ALL TOGETHER for an EXPERIENTIAL GOAL

Action:

Explore as necessary to answer the discussion questions.

Discuss:

How do the different feedback modes (fade vs. on/off and small vs. large lines) affect *your movement* and *how you relate* to the system? Is the physical world (your actions) changed because of the feedback? How could you imagine using this system in practice? What do you think it was used for (hint – this was a real project built in 2010)

If you were to propose your own system using frame differencing as motion capture, what would you propose? Make sure to discuss your experiential goal, types of feedback, intended audience, and intended primary location for your system. Be imaginative and creative!

what about PROCESSING?

Action: Open the processing sketch and run it.

Discuss: What part of the experiential media system model is missing? Why? Are you sure?

Action (again):

Look at the code. Try to:

- Find where the threshold is defined and change it. Try several different numbers and see what happens.
- In void setup(), change the size of the window. Try it many different ways. See if you can get it to fit your entire computer screen. How did you do that? (Hint "esp" will exit the sketch if it gets too big where you can't close it).
- change the color of the black pixels to green. Change them to another color of your choice.

So, What is my HOMEWORK?

Your homework is to individually respond to six reflection questions regarding the activities in class today. The questions can be found under “Projects and Assignments” on Blackboard, called “Frame Differencing Reflection Homework.” They can also be seen on the following slide.

It is due **Tuesday, October 15th, at 11:59pm.**

If you have a friend who was absent today, help them out - let them know that they can still get credit for this homework assignment if they turn it in on time, but they might need some help exploring the frame differencing code.

This assignment will be graded for quality, not completion - so actually put some effort into it.

Homework:

For the following questions, refer to both the MaxMSP frame differencing patch and the Processing frame differencing sketch shown in class on October 9th and 10th.

1. What was the sensor in both systems?
2. Is frame differencing sensing, analysis, or feedback?
3. In 2-5 sentences, describe the process of “frame differencing” and how it is used as motion capture.
4. Describe exactly what the world “threshold” meant in both systems.
5. How do the different feedback modes in the Max patch (fade vs. on/off and small vs. large lines) affect *your movement* and *how you relate* to the system? Is the physical world (your actions) changed because of the feedback? If you were to change the feedback, how would you change it?
6. In 4-5 sentences, imagine and describe an interactive system as we've defined it in class with sensing, analysis, feedback and an experiential goal. Tell its context; where it would be used and who would likely use it. Be creative and think outside the box.
7. Google your idea and find similar existing media. Post a link to and short description of a video or article that demonstrates a system similar to the one you just described.

Cover

Download

Exhibition

Reference
Libraries
Tools
Environment

Tutorials
Examples
Books

Overview
People
Foundation

Shop

» Forum
» GitHub
» Issues
» Wiki
» FAQ
» Twitter
» Facebook

Overview. A short introduction to the Processing software and projects from the community.

We're thrilled to announce the launch of Processing 2.0—the latest incarnation of the programming language, development environment, and online community that has grown dramatically since its debut in 2001. This new release builds on the versions of Processing that have been downloaded almost two million times. We invite you to download Processing 2.0 from www.processing.org. The Processing software is free and open source, and runs on the Mac, Windows, and GNU/Linux platforms.

For the past twelve years, Processing has promoted software literacy, particularly within the visual arts, and visual literacy within technology. Initially created to serve as a software sketchbook and to teach programming fundamentals within a visual context, Processing has also evolved into a development tool for professionals. We stand by our mission statement:

Processing seeks to ruin the careers of talented designers by tempting them away from their usual tools and into the world of programming and computation. Similarly, the project is designed to turn engineers and computer scientists to less gainful employment as artists and designers.

The Processing 2.0 release focuses on faster graphics, new infrastructure for working with data, and enhanced video playback and capture. It also expands the potential of the programming environment. The new Modes feature allows other programming systems, such as JavaScript and Android, to be easily used from within the development environment. The new Contributions Manager makes it simple to distribute and install extensions developed by the community. The P2D and P3D renderers are now built using modern OpenGL, and programs can now utilize custom GLSL shaders.

Processing continues to be an alternative to proprietary software tools with restrictive and expensive licenses, making it accessible to schools and individual students. Its open source status encourages the community participation and collaboration that is vital to Processing’s growth. Contributors share programs, contribute code, and build libraries, tools, and modes to extend the possibilities of the software. The Processing community has written more than a hundred libraries to facilitate computer vision, data visualization, music composition, networking, 3D file exporting, and programming electronics. Please read about our amazing team of volunteers and the list of community contributions at www.processing.org.

Processing is currently developed primarily in Boston (at Fathom Information Design), Los Angeles (at the UCLA Arts Software Studio), and New York City (at NYU’s ITP).

Education

From the beginning, Processing was designed as a first programming language. It was inspired by earlier languages like BASIC and Logo, as well as our experiences as students and teaching visual arts foundation curricula. The same elements taught in a beginning high school or university computer science class are taught through Processing, but with a different emphasis. Processing is geared toward creating visual, interactive media, so the first programs start with drawing. Students new to programming find it incredibly satisfying to make something appear on their screen within moments of using the software. This motivating curriculum has proved successful for leading design, art, and architecture students into programming and for engaging the wider student body in general computer science classes.

Processing is used in classrooms worldwide, often in art schools and visual arts programs in universities, but it's also found frequently in high schools, computer science programs, and humanities curricula. Museums such as the Exploratorium in San Francisco use Processing to develop their exhibitions. In a National Science Foundation-sponsored survey, students in a college-level introductory computing course taught with Processing at Bryn Mawr College said college-level introductory computing course taught with Processing at Bryn Mawr College said they would be twice as likely to take another computer science class as the students in a class with a more traditional curriculum.

The innovations in teaching through Processing have been adapted for the Khan Academy computer science tutorials, offered online for free. The tutorials begin with drawing, using most of the Processing functions for drawing. The Processing approach has also been applied to electronics through the Arduino and Wiring projects. Arduino uses a syntax inspired by that used with Processing, and continues to use a modified version of the Processing programming environment to make it easier for students to learn how to program robots and countless other electronics projects.

Culture

The Processing software is used by thousands of visual designers, artists, and architects to create their works. Projects created with Processing have been featured at the Museum of Modern Art in New York, the Victoria and Albert Museum in London, the Centre Georges Pompidou in Paris, and many other prominent venues. Processing is used to create projected stage designs for dance and music performances; to generate images for music videos and film; to export images for posters, magazines, and books; and to create interactive installations in galleries, in museums, and on the street. Some prominent projects include the House of Cards video for Radiohead, the MIT Media Lab’s generative logo, and the Chronograph projected software mural for the Frank Gehry-designed New World Center in Miami. But the most important thing about Processing and culture is not high-profile results – it's how the software has engaged a new generation of visual artists to consider programming as an essential part of their creative practice.

Research

Software prototyping and data visualization are two of the most important areas for Processing developers. Research labs inside technology companies like Google and Intel have used Processing for prototyping new interfaces and services. Companies including General

Electric, Nokia, and Yahoo! have used Processing to visualize their internal data. For example, the New York Times Company R&D Lab used Processing to visualize the way their news stories travel through social media. The NSF and NOAA supported research exploring phytoplankton and zooplankton diversity that was realized at the University of Washington as a dynamic ecology simulation. Researchers at the Texas Advanced Computer Center at UT Austin have used Processing to display large data visualizations across a grid of screens in the service of humanities research.

Foundation

With the launch of Processing 2.0, we are excited to take a new step forward and establish the Processing Foundation. Up to this point, Processing has been developed almost exclusively by volunteers. We realized that funding was essential to support Processing’s vast user base and maintain the high quality of the software.

The primary charge of the Foundation is to develop and distribute the Processing software, both the core Application Programming Interface (API) and the programming environment, the Processing Development Environment (PDE). To meet this charge, we welcome donations from individuals and organizations to assist in developing the Processing software. The board of directors for the Processing Foundation consists of Ben Fry, Casey Reas, and Daniel Shiffman. The first two members of the board of advisors are John Maeda (President, RISD) and Red Burns (Founder, NYU ITP). We are in the final stages of applying for non-profit status as a 501(c)(3) organization.

History

Processing was started by Ben Fry and Casey Reas in the spring of 2001,while both were graduate students at the MIT Media Lab within John Maeda's Aesthetics and Computation research group. Development continued in their free time while Casey pursued his artistic and teaching career and Ben pursued a Ph.D. and founded Fathom Information Design. Many of the ideas in Processing go back to Muriel Cooper's Visual Language Workshop, and it grew directly out of Maeda's Design By Numbers project, developed at the Media Lab and released in 1999. The Wiring and Arduino projects, in turn, grew out of Processing while Casey was teaching at the Interaction Design Institute Ivrea in Italy. Processing also prompted John Resig (jQuery) to build Processing.js, a JavaScript version that then inspired more related work such as the Khan Academy curriculum in computer science. Versions of Processing that use Python, Ruby, ActionScript, and Scala are also in development. Processing and its sister projects have inspired over twenty educational books.

Links

Processing http://processing.org
Fathom Information Design http://fathom.info
UCLA Arts Software Studio http://software.arts.ucla.edu

NYU ITP http://itp.nyu.edu

For more information, please write to foundation@processing.org

Processing was initiated by Ben Fry and Casey Reas. It is developed by a small team of volunteers.

Thanks for your help. We've received your bug report.

(http://cycling74.com/)

Try Max for Free (/downloads/)

Max

Max gives you all the tools to create your own music, sound, video, and interactive media applications. You can arrange boxes on a canvas and connect them together to create, experiment, and play.

Discover and connect as you build.

Explore sounds, images, and interfaces in new combinations. Building in Max is a fluid, visual, and immediate process. Instantly see and hear the results with every change you make.
Get details ➔ (/products/max/visual-programming/)

Andrew Spitz: Interaction Designer

"I feel like Max is such a good tool for the people that don’t know much about programming and yet still want to have control over the code for their project."
Hear more from Andrew ➔ (/2012/10/29/an-interview-with-andrew-spitz/)

Expand your music and sound creations without rules.

Build your own world of sound using sample-manipulation, synthesis tools, high-quality filters, spectral processing, real-time recording and playback. Process, resample, slice, and modulate everything in any combination.
Get details ➔ (/products/max/audio-music/)

Robert Henke: Audiovisual Artist and Developer

Robert writes music, makes custom Max for Live devices, and creates installations.
"[Max] has the potential to change the way you think about music. Total liberation."
Hear more from Robert ➔ (/2009/11/21/an-interview-with-robert-henke/)

Create unique visual experiences.

Control visuals with audio, MIDI, or any other data source. Integrate vector graphics, compositing effects, color manipulations, and videos.
Get details ➔ (/products/max/video-jitter/)

Sue-C: Handmade Visuals Artist and Performer

SUE-C uses Max for her live performances and recorded work to animate hundreds of different objects. "There has never been anything Max was unwilling to let me try."
Hear more from Sue-C ➔ (/2013/02/28/an-interview-with-sue-c/)

Generate stunning 3D graphics.

Create your own 3D worlds, complete with lighting, materials, and multiple cameras. Animate these 3D models in real-time, combine 3D rendering with shader effects, and use the advanced physics engine for animation control.
Get details ➔ (/products/max/3d-graphics/)

Masato Tsutsui: Video Artist and AV Technician

"Want to use another? Everyone does that, but we absolutely can't ignore Max and its important part."
Learn more about Masato ➔ (http://adsr.jp/)

Connect sensors and electronics to create interactive media.

With its ability to communicate with other applications and hardware, Max is the glue that holds your project together.
Get details ➔ (/products/max/physical-computing/)

Ali Momeni: Builder and Experience Designer

"Max is ever-more-forcefully breaking out of electronic music and becoming an attractive instrument for all manners of creative activity."
Hear more from Ali ➔ (/2010/05/25/an-interview-with-ali-momeni/)

Combine controllers, music, lighting, & visuals.

Max has a full suite of tools that allow you to connect music, video, control, and lighting systems together. Integrate standard MIDI or OSC controller devices, or connect custom hardware to control a show unlike any other.
Get details ➔ (/products/max/show-control/)

Ryan Holsopple of 31 Down: Performance

"Max is the great translator that allows us to speak multiple protocols simultaneously and creates the most versatile control computer, making the programming as creative as the performance."
Hear more from Ryan ➔ (/2013/06/10/an-interview-with-ryan-holsopple/)

Our users matter.

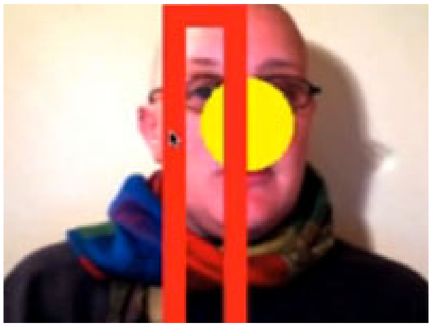
We are inspired every day by the creativity of the performers, artists, composers and scientists who use Max.

Daito Manabe

Sound Artist and Experimentalist

Uses Max to make unusual commercials and jaw-dropping experiments with electronics.

(http://www.daito.ws/en)



Pauline Oliveros

Composer and Humanitarian

The Adaptive Use Project strives to bring music-making to people for the first time.

(/2007/12/07/the-adaptive-use-instruments-project/)

Barney Haynes

Media Artist and Educator

Inspires students to combine new and scavenged technology to create interactive experiences.

(/2005/09/28/a-video-interview-with-barney-haynes-media-artist-educator/)

Elise Baldwin

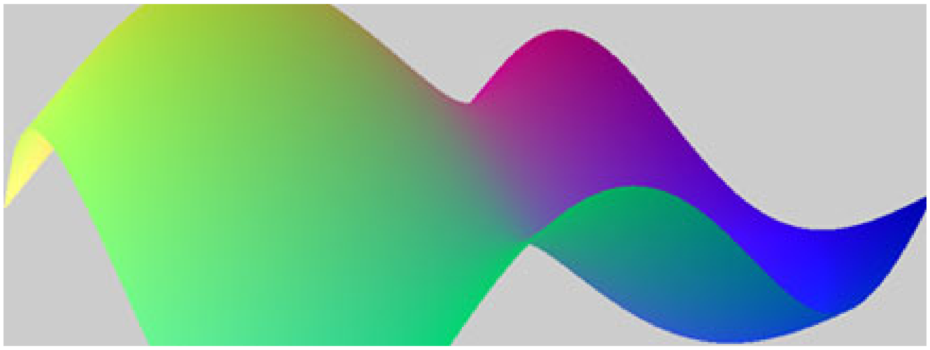
Intermedia Artist and Audio Director

Taps her creativity in prototyping and artwork.

(/2010/04/23/an-interview-with-elise-baldwin/)

Extend Max

Add-ons to make your projects exceed their potential.



GEN

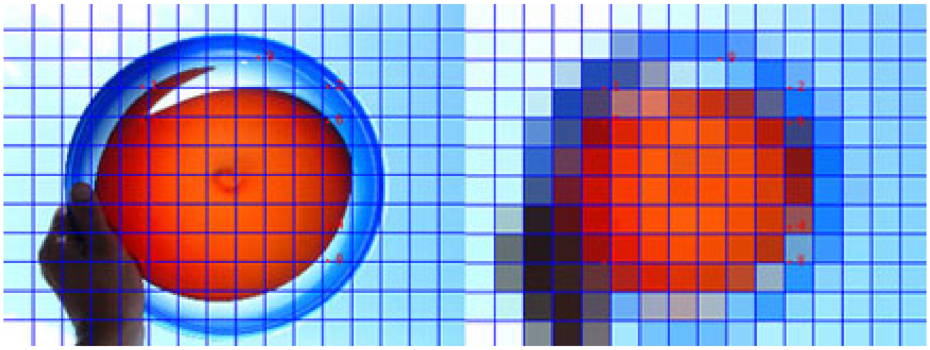
Visually develop efficient, platform-independent code.

(/products/gen/)

MIRA

An iPad app that automatically connects to Max and mirrors your interface.

(/products/mira/)



CYCLOPS

Analyze and track live video in real time.

(/products/cyclops/)
Try Max for 30 Days (/downloads/)

Watch Max User Videos (/videos/)

Take a Tour of Max (/videos/product/)

- Products (http://cycling74.com/products/)
- Max (http://cycling74.com/products/max/)
- Gen (http://cycling74.com/products/gen/)
- Max For Live (http://cycling74.com/products/maxforlive/)
- Soundflower (http://cycling74.com/products/soundflower/)
- Cyclops (http://cycling74.com/products/cyclops/)
- M (http://cycling74.com/products/m/)
- C74 Music Label (http://cycling74.com/products/c74music/)
- Downloads (http://cycling74.com/downloads/)
- Max (http://cycling74.com/downloads/)
- Soundflower (http://cycling74.com/products/soundflower/)
- M (http://cycling74.com/downloads/)
- Discontinued Products (http://cycling74.com/downloads/discontinued/)
- Users (http://cycling74.com/community/)
- Forums (/forums)
- Projects (/project)
- Toolbox (/toolbox)
- Max Wiki (http://cycling74.com/wiki/)
- Interviews (http://cycling74.com/category/articles/interview/)
- Articles (http://cycling74.com/category/articles/)
- Schools That Teach Max (http://cycling74.com/wiki/index.php?title=Category:PeoplePlaces)
- Support (http://cycling74.com/support/)
- Forums (/forums/)
- Authorization (http://cycling74.com/support/authorization/)

- FAQ (http://cycling74.com/support/faq/)
- Did You Know? (http://cycling74.com/category/articles/did-you-know/)
- Tutorials (http://cycling74.com/category/articles/tutorials/)
- Documentation & Books (http://cycling74.com/wiki/index.php?title=Max_Documentation_and_Resources)
- Company (http://cycling74.com/company/)
- About Us & Contact (/company/)
- Newsletter (http://cycling74.com/company/newsletter/)
- Press Releases (http://cycling74.com/category/company/press-releases/)
- Facebook (http://facebook.com/cycling74)
- Twitter (http://twitter.com/cycling74)
- YouTube Channel (http://www.youtube.com/user/cycling74com)
- Google+ (https://plus.google.com/100536393816379595897)

Subscribe to our newsletter

Let us tell you about notable Max projects, obscure facts, and creative media artists of all kinds.

email address

Subscribe

Lecture 2A - Audio Recording Techniques (10 minutes)

Objective(s):

1. *Given the description of a complex human activity, students will justify their selection of recording devices and techniques.*
2. *Students will employ appropriate recording techniques to capture human activity.*
3. *Students will choose a media-editing tool based on an experiential goal, and justify their choice in a short paragraph.*

#	Type	Script	Media Assets	Notes
1	title	Welcome to our next lecture, one that discusses some practical skills for recording media. We'll discuss the process of recording audio and how to get the best possible source material.		Display course title and "Audio Recording Techniques"
2	conte nt	With the proliferation of accessible recording devices embedded in phones and laptop computers, many of you have probably recorded some audio at some point already, so you already have a handle on how it works, but let's review some specifics.		
3		Basics of Sound All sound is vibration. To break it down, every sound requires 4 things - energy, vibration, medium of transmission, and a receiver. A source puts out a vibration, which moves the air molecules surrounding the source; and just like dropping a rock in a pond, the air ripples with vibrations until it hits something, like your ears. The human ear is capable of detecting the mine-ute change in air pressure; the ear drum is a stretched membrane that vibrates at the same frequency as the sound waves that are travelling through the ear canal. Tiny hair cells called cilia in the inner ear respond to the vibrations from the ear drum, and these hair cells turn physical vibrations into electric nerve impulses. Then some magic happens – meaning this part of the process is not very well understood, but		Display: Basics of Sound - All sound is vibration - All sound requires 4 things: energy vibration medium of transmission (air) and a receiver

		the auditory cortex can now interpret the nerve impulses into the gamut of complexities we call sound.		
4		How do microphones work? If sound hits a microphone, something similar to the process in the ear happens - changes in pressure vibrate a thin membrane called the diaphragm, which is attached to a coil, and the movement of the diaphragm converts creates an electrical signal in which the voltage and current are proportional to the original sound.	**dynamic_mic_diagram.jpg** http://ohda.matrix.msu.edu/wp-content/uploads/2012/06/micdiagram2922.jpg	display text: “How Do Microphones Work?” along with the image/
5		Ok, So - you want to make a recording. Where do you start? First, you’ll need some gear - a microphone, and an audio recorder of some sort. There are two basic types of microphones - condenser and Dynamic. Some of the pros and cons are listed here. Condenser microphones are much better for recording - they reproduce your frequencies more faithfully, with more clarity and definition. They are better for acoustic instruments. But of course, this comes with sacrifice, they break more easily and require a preamp (electricity). Dynamic Microphones generally have a more varied frequency response, which means the sound reproduction is not as faithful - which is not necessarily a bad thing. Some singers really like certain microphone set-ups because they reproduce their voice in a way that sounds better than their original. They are typically cheaper and more rugged, meaning they are better for stage performing and travel. *<don’t read this!>* *Condenser* • *Reproduces frequencies faithfully = truer to the actual sound, more clarity* • *Better for acoustic instruments (soft vocals, acoustic*	**condenser_mic.jpg** **dynamic_mic.jpg** (both are from http://www.musiciansfriend.com/)	Title text: Types of Microphones. Content text is to the left in italics. Place images along with their respective names.

		guitar) • *Easily breakable* • *requires a preamp (electricity)* *Dynamic* • *Varied frequency response = less true to original sound* • *More rugged, better for stage and travel*		
6		Let's say you're making a podcast that explains the difference in sounds between several different kinds of shakers. Since you're at your desk and you'd prefer a true sound because of the subtle nature of your topic, you choose a condenser mic. Now... Where do you place the microphone?	various_shakers.jpg (http://www.philtulga.com/products-shakers.jpg) *Let me know if the quality of this image is not good enough.*	
7		Microphone placement depends on what you are recording. In general, close but not too close. For solo speaking voices, the microphone should be a few inches away; not so close that loud "p's" create a pop from the burst of air coming from the lips. You can use a pop filter (basically a foam covering of the microphone) to reduce these. Pay attention to the space you are recording in; you will pick up reverberations from the space the further away you the microphone is from the subject, which can be an advantage or disadvantage, depending on what you want! Back to making that percussion podcast - In this case, you'll place the microphone a few inches away from yourself when you are speaking, and slightly further for the shaker. For both of these sound sources though, you'll need to test and then set your levels. *<don't read this!>* *Microphone placement* • *depends on context - what are you recording?*		text only. Display bullets from left

		• *Generally, have the microphone close to the subject* ○ *But far enough from a speaker to avoid “pops” from hard spoken consonants.* ○ *The closer the sound, the more true it will be, and the less impacted by space.*		
8		*Audio Levels* How do I set my levels? Just like Goldilocks - not too low, not too high - just right. If it is too low (somewhere below -10dB, you may not be able to hear the subject clearly). If it is too high, you’ll get clipping and distortion of the sound. The best level is when your loudest sound is just below clipping. What’s clipping? Clipping is when your sound is so loud that it goes above the limit the microphone can process. In analog, this means there is a voltage limit, and in digital, an integer limit. When the processor can’t handle the sound, it clips it at the top, which distorts the audio. Even if you process the sound after it has been recorded to make it quieter, the distortion remains because the audio has already been compromised. On the sound meter shown here, clipping happens at 0.0, which represents the highest possible sample value. Next to it, in the blue waveform, you can see on the top right, the blue waveform reaches the top and is cut off. This is a visual representation of clipping. Avoid clipping by recording a test of the subject while watching the level meter and adjusting the input volume on the device. Any devices or software programs you record with will have a meter similar to the ones seen here and a setting that allows you to adjust the intake volume for your microphone.	**MeterBridge.png** http://www.faberacoustical.com **2A_waveform_clipping.png**	Display two images, as well as text: For best results, the loudest sound should be just below clipping.
9		Noise. Any sound that you record that was not your intended subject is considered noise. Conveniently, in other types of signal processing other than sound such as electromagnetic radiation, image processing, and sensor data, you can use the same term, noise, to refer to any information you collected that isn’t relevant to your		Title: Noise

		intended observation. This will become relevant if you build any interactive system using some type of sensor like the accelerometer in the wiimote. But, back to sound.		
10		Noise is typically not desired in a quality recording. Background sound can certainly be useful if it is designed, for example, if you are designing the sound for a restaurant scene in a movie, you’d intentionally and carefully craft the background sound to be authentic, but not interfere with the foreground sound. But this is different from background noise, which is that which was not intended - usually a hum, buzz, or whispery sound, and you’ll want to know how to minimize background noise. It comes from two places, 1. the noise of the actual room and 2., the device itself - the electronics inside can create a hum in the recording that didn’t exist in the space. You can avoid this by getting a good microphone.		Title: “Noise” display: “comes from 2 sources: (1) actual background noise in the space (2) the device itself
11		Your “noise floor” is the amplitude (remember, like loudness) of the rest of your recording other than your intended subject. So if you are making your podcast in your living room, the subject is your speaking voice, and the sound of the instruments. The background noise is a car driving by, a television in the other room, the fan or air conditioner in the room, or you shuffling on the couch. In this image, the exact same clip is shown twice, on the left, the original recording, on the right, processed to remove the noise. You’ll notice that the “noise floor” looks like a floor along the 0.0 line. In this case the editing worked like a dream, but that’s because I created the example using a noise generator, so it was pretty cut and dry. In most cases, noise in a scene is much more complex, and while noise removal in post production does help, it can only do so much. One good practice is to record a few seconds of just room noise, without any subject noise. Most audio editors can use this sample of room noise as part of the post production noise removal process.	**2A_waveform_noisecomparison.png**	“noise floor: “the amplitude of the rest of your recording other than your intended subject.” show image on screen for the whole time.

12		Certainly there are obvious ways to reduce noise in this example; ask your roommate to turn off the TV for a moment, make sure doors and windows are closed, turn off the fan or A/C for the duration of the recording session, and be careful of your own actions physical actions, like shifting in your chair. Less obvious ways include moving away from objects that have hums, like the fan on your computer or the refrigerator, recording in a place that has a lot of cloth, which will absorb ambient sounds, and getting the microphone as close to the source as possible, and of course – using good microphones if you can get them. *<don’t read this!>* • *turn off fans, air conditioners, move away from refrigerators and other things with hums.* • *record in a quiet place that has a lot of cloth to absorb ambient sound* • *get the microphone as close the source as possible.* • *use quality microphones* *<read again>* You are always going to have some noise, but you want to take as many precautions as you can to make sure your noise floor is as low as possible.		text only; Content to display: “Minimize noise by: “ *and bullets to the left.*
13		One last tip - If you need to sync audio and video, you can clap into the microphone and video camera at the same time - later, during the editing process, you’ll be able to see the amplitude spike on the audio waveform and match it up exactly when your hands go together on the video.		text: Pro tip: Syncing Audio with Video
14		That’s your rundown on capturing audio! Editing is where the magic happens, but you have to start with good source audio to end up with a good product.		