

---

# CSCI 1050

## Introduction to Computer Science: Multimedia

### Contents

<b>1</b>	<b>Overview</b>	<b>1</b>
1.1	Catalog Description . . . . .	1
1.2	Detailed Description . . . . .	2
<b>2</b>	<b>Course Administration</b>	<b>2</b>
2.1	The Staff . . . . .	2
2.2	Class Meetings . . . . .	2
2.3	Textbook . . . . .	3
<b>3</b>	<b>Online Resources</b>	<b>3</b>
3.1	CSCI 1050 Web Page: <a href="http://cs.slu.edu/~goldwasser/1050/">http://cs.slu.edu/~goldwasser/1050/</a> . . . . .	3
3.2	Electronic Assignment Submission . . . . .	3
3.3	Email with Instructor . . . . .	4
<b>4</b>	<b>Flipped Classroom Style</b>	<b>4</b>
<b>5</b>	<b>Course Grades</b>	<b>5</b>
5.1	Graded Work . . . . .	5
5.2	Course Grades . . . . .	5
5.3	Late Policy . . . . .	6
5.4	Academic Integrity . . . . .	6
<b>6</b>	<b>Additional Information</b>	<b>7</b>
6.1	Supporting Student Success . . . . .	7
6.2	Tutoring Resources at SLU . . . . .	8
6.3	Computing Resources at SLU . . . . .	8

## 1 Overview

### 1.1 Catalog Description

“An introduction to computer programming motivated by the creation and manipulation of images, animations, and audio. Traditional software development concepts, such as data representation and control flow, are introduced for the purpose of image processing, data visualization, and the synthesis and editing of audio.”

## 1.2 Detailed Description

As an introduction to the field of computer science, we will explore an interesting dichotomy of precision and creativity, which feeds the development of new tools, algorithms, and data analyses. The precision and attention to detail is required because as “powerful” as computing technology seems, the machines simply follow the instructions they are given, and if they are given flawed instructions, then they will not behave as desired. However, computer science goes well beyond mere coding. Creativity allows us to envision new uses for computing, and to develop new tools or find new designs that improve on existing technologies.

In this course, we further emphasize the creative process of software design by focusing on the creation of static, animated, and interactive visualizations, as well as the manipulation of existing imagery and audio. We will use the Processing programming language (more specifically, version Processing 3, released in 2015). Processing is an open-source programming language that was originally developed in the MIT Media Lab as a crossover between graphic design and computer programming. Additional references about the language can be found in the “documentation” section of the course web page.

## 2 Course Administration

### 2.1 The Staff

Instructor: Dr. Michael Goldwasser  
Email: [goldwamh@slu.edu](mailto:goldwamh@slu.edu)  
Web: <http://cs.slu.edu/~goldwasser>  
Office: Ritter Hall 355  
Telephone: (314) 977-7039  
Office hours: Tuesdays 10:00–10:50am  
Wednesdays 2:00–3:00pm  
Thursdays 12:30–1:30pm  
or by appointment

### 2.2 Class Meetings

The material will be presented in two weekly meetings, with mandatory attendance. We discuss the classroom style later in this document. Information on the daily topics can be found on the course schedule web page.

Time: Tue/Thu, 11:00am–12:15pm  
Place: Ritter Hall 236

## 2.3 Textbook

The required textbook for this course is:

Title: *Getting Started with Processing: A Hands-On Introduction to Making Interactive Graphics*  
Authors: Casey Reas and Ben Fry  
Publisher: Maker Media Inc. and O'Reilly, 2015  
ISBN: 978-1-457-18708-7  
Website: [shop.oreilly.com/product/063920031406.do](http://shop.oreilly.com/product/063920031406.do)

This text is authored by the two original creators of Processing and it is mirrored by some of the online documentation at [processing.org](http://processing.org). I find it a very succinct (and affordable) desk reference for the language.

For those looking for a much more in-depth book (with significantly more emphasis on visual art), these same authors have a book titled *Processing: A Programming Handbook for Visual Designers and Artists, Second Edition*, published by MIT Press.

Finally, another wonderful book is *Learning Processing: A Beginner's Guide to Programming Images, Animations, and Interaction, Second Edition* by Daniel Shiffman. Although we will not follow the table of contents in quite the same order, we will be relying on many of Daniel's online videos as a resource. If you like those, you are welcome to take a look at his book as well.

## 3 Online Resources

### 3.1 CSCI 1050 Web Page: <http://cs.slu.edu/~goldwasser/1050/>

With the exception of the first day's printed handouts, most of the information for this course will be distributed only by means of the course web page. This web site will contain all assignments, a schedule of lectures, examples from lecture, and links to many other sources of information.

The web page contains some information (e.g. solutions, submitted assignments, individual grades) that is more sensitive and therefore which will be available to students in the class only after they have identified themselves properly. To gain access to these parts of the web page, a student must first complete an online questionnaire, creating a unique identity and password.

### 3.2 Electronic Assignment Submission

Programming projects for this course must be submitted electronically! The submission procedure will be done through the course web page, and allows students to submit from any computer connected to the Internet. Each student in this class will be selecting a unique username/password combination solely for use in identifying the student when using the course web page. Details of the procedure are discussed at:

<http://cs.slu.edu/~goldwasser/1050/submit/>

### 3.3 Email with Instructor

Face-to-face contact in class and in office hours is most desirable. Yet email is a convenient form of communication as well. I try to respond to email promptly, including at least once each evening when possible.

If your question involves your progress on a current programming assignment, my response will be more informative if you can point out the specific problem you have encountered, and if I am given a copy of your entire source code.

## 4 Flipped Classroom Style

We will primarily be using a flipped classroom style in which students learn more basic information prior to class times through a combination of learning materials (book, videos, webpages), and in which formal class times are devoted to more active learning experiences.

During the first half of the semester, we will adhere to the following workflow surrounding each knowledge unit:

- Prior to the class meeting, a summary of the knowledge unit will be posted on the course schedule, including a list of new topics to learn, associated chapters from the textbook, and links to other relevant websites, tutorials, and videos. The summary will also contain materials for use during the class meeting
- At the beginning of the class meeting, students will be randomly assigned to work in pairs.
- Pairs will work together on a series of “challenges” posted to the unit summary prior to class. Students will not be graded on these challenges, nor will the work be submitted. Pairs can choose which challenges to attempt and we will intentionally provide far more challenges than any single pair would be likely to finish during the official meeting time.
- When 10 minutes remain in the meeting time (i.e., at 12:05pm), a written quiz will be distributed to the class. Students will work in their pairs to complete the quiz, with each pair submitting a single response. Students will be allowed to use Processing on the classroom computers to develop or test their solutions when completing the quiz.

During the second half of the semester, we may devote some meetings to more of a studio session, with students able to work on larger projects that will extend across several class periods.

## 5 Course Grades

### 5.1 Graded Work

Course grades will be based on the following components:

- **Quizzes (25%)**

We will likely have between 15 and 20 quizzes (depending on our use of class time during the second half of the semester). To provide some flexibility for absences or poor days, we will drop your lowest two quiz scores at the end of the semester, and average the rest of them for this component of your course grade.

- **Midterm Exam (25%)**

A midterm exam will be given during the entirety of class on Thursday, 3 March 2016.

- **Projects (50%)**

Projects will consist of more significant Processing sketches than those done as in-class activities. Although we may devote some working time in class for these, they will primarily be done outside of class. We will likely have 5 regular projects weighted equally (although this is subject to change), and a final project that is worth twice the weight of a regular project.

### 5.2 Course Grades

Letter grades will be based on each students overall percentage of awarded points according to the following formula.

Student percentage above 90% will result in a grade of A or better.

Student percentage above 87% will result in a grade of A- or better.

Student percentage above 83% will result in a grade of B+ or better.

Student percentage above 80% will result in a grade of B or better.

Student percentage above 77% will result in a grade of B- or better.

Student percentage above 73% will result in a grade of C+ or better.

Student percentage above 70% will result in a grade of C or better.

Student percentage above 67% will result in a grade of C- or better.

Student percentage above 60% will result in a grade of D or better.

Student percentage below 60% will result in a grade of F.

Any modification to this scale at the end of the year will be *in favor of the students*. That is we may later decide to award an A to a student who is slightly below the above cutoff, but we certainly will not deny an A from someone who is above the cutoff.

### 5.3 Late Policy

All exams and quizzes must be taken promptly at the required time. Requests for rescheduling an exam will only be considered if the request is made prior to the start of the exam, or else in an “emergency” situation with appropriate documentation.

For projects, we wish to allow students to continue to work comfortably beyond the official deadline when a little more time will result in more progress, while at the same time discourage students from falling significantly behind pace and jeopardizing their success on future projects. Our solution is the following exponentially decaying late formula.

We will consider a project submission “complete” when any part of the assignment is last submitted or modified. Any assignment that is not complete promptly by its due date and time will be assessed a penalty based on the formula  $S = R \cdot e^{-h/173}$ , where  $S$  is the grade given,  $R$  is the grade the work would have received had it been turned in on time, and  $h$  is the amount of time (in hours or fractions thereof) that the work was late. Examples:

- work turned in 1 hour late receives over 99.6% of its original credit
- work turned in 5 hours late receives over 97% credit
- work turned in one full day late receives less than 88%
- work turned in two full days late receives less than 76%
- work turned in five days late receives less than 50%

The above policies will be waived only in an “emergency” situation with appropriate documentation.

### 5.4 Academic Integrity

Academic integrity is honesty, truthful and responsible conduct in all academic endeavors. The mission of Saint Louis University is “the pursuit of truth for the greater glory of God and for the service of humanity.” Accordingly, all acts of falsehood demean and compromise the corporate endeavors of teaching, research, health care, and community service via which SLU embodies its mission. The University strives to prepare students for lives of personal and professional integrity, and therefore regards all breaches of academic integrity as matters of serious concern.

The governing University-level Academic Integrity Policy can be accessed on the Provost’s Office website. A more detailed policy statement is given by the College of Arts & Science ([www.slu.edu/colleges/AS/academic\\_honesty.html](http://www.slu.edu/colleges/AS/academic_honesty.html)), also applying to this course. In addition to those general statements, we wish to discuss our policy in the context of this course. When it comes to learning and understanding the **general course material**, you may certainly use other reference materials and you may have discussions with other students in this class or other people from outside of this class. This openness pertains to material from the text, practice problems, general syntax and use of the C++ language or other computing tools.

However, when it comes to **work that is submitted for this course**, you are not to use or to search for any direct or indirect assistance from unauthorized sources, including but not limited to:

- other students in this class
- past students, whether from this school or other schools
- other acquaintances
- other texts or books
- online information other than that referenced by course materials
- Processing projects found online

Acceptable sources of information include consultations with the instructor, teaching assistants, or members of organized tutoring centers on campus, as well as any materials explicitly authorized in a project description. Even in these cases, if you receive significant help you should make sure to document both the source of the help as well as the extent.

On certain programming projects, we will explicitly allow students to work in pairs. In this case, conversations between partners is both permissible and required. Furthermore, both students are expected to contribute significantly to the development of the submitted work. It is unethical to allow a partner to “sign on” to a submission if that partner did not significantly contribute to the work.

Any violations of these policies will be dealt with seriously. Penalties will apply as well to a student who is aiding another student. Any such violations will result in a minimum penalty of a zero on the given assignment that cannot be dropped, and severe or repeated violations will result in an immediate failing grade in the course. Furthermore all incidents will be reported in writing to the Department and/or the Dean, as per the College procedure.

## 6 Additional Information

### 6.1 Supporting Student Success

In recognition that people learn in a variety of ways and that learning is influenced by multiple factors (e.g., prior experience, study skills, learning disability), resources to support student success are available on campus. Students who think they might benefit from these resources can find out more about:

- Course-level support in the remainder of this section or by asking the instructor
- University-level support (e.g., tutoring/writing services, Disability Services) by visiting the Student Success Center (BSC 331) or by going to [www.slu.edu/success](http://www.slu.edu/success).

Students with a documented disability who wish to request academic accommodations are encouraged to contact Disability Services at 314-977-3484 or visit the Student Success Center. Confidentiality will be observed in all inquiries.

## 6.2 Tutoring Resources at SLU

Our department employs many junior/senior computer science majors to help out in our department labs. Those students are also available to provide assistance with course materials at such times.

Our department web page maintains a current list of the available times and locations at [cs.slu.edu/undergrad-cs/lab-hours](http://cs.slu.edu/undergrad-cs/lab-hours).

As stated in the Academic Integrity policy of Section 5.4, these tutors are an acceptable resource for help, yet you should still document both the source of the help as well as the extent, if significant.

## 6.3 Computing Resources at SLU

Processing is installed in Ritter 323, however that room is not generally open and available outside of classtime, nor can files saved to their hard drives be accessed elsewhere. Fortunately, Processing is freely available software for all major computing platforms and can be installed elsewhere. We recommend that you install it on any machine you have, and that you save any in-class work either on a flash drive or on your personal cloud storage.

Our department runs a Linux computer server named `turing` that serves as the primary computing environment for many of our other courses, and a lab in Ritter 117. An account will be created for you to use that system (in person or remotely) and Processing will be installed there as well. If you are interested in more information about our facilities. See [cs.slu.edu/computing-resources](http://cs.slu.edu/computing-resources) for further documentation regarding use of `turing`'s facilities.