Programming for Engineers -- Elective

Course Description

This course introduces students to the software development process in a variety of settings, so that they acquire a wide set of programming skills. The first half of the course uses the P5.js library and the JavaScript language, while the second half uses Python. Emphasis on robust program design and sound software engineering practices will be put throughout the course. Each instructional unit will take 3-4 weeks.

Course Objectives

- Write programs using variables, functions, recursion, conditionals, loops, lists, arrays and objects.
- Organize code hierarchically and make use of modules.
- Design, implement and debug programs of several hundred lines of code, spanning several files.
- Demonstrate familiarity with the functions in the standard library of one or more languages.
- Explain the basic structure and organization of a medium-size program.
- Make reasonable choices for algorithms and data structures when designing a program.
- Properly use documentation and online resources to resolve programming questions.
- Work efficiently in a team to build programs collaboratively.

Assessing Performance

Formative assessment includes worksheets and several practice activities for each lesson, and unit quizzes. Summative assessment includes a programming project at the end of each unit.

Course Essentials

Equipment	**Cost/Unit**
Reusable material for physical computing	$1,500
Classroom set of computers	$0 if you already have some, $500-600 per computer if you need to purchase
Software	All the software used in this course is open source and widely available free of charge, but installation of software in local computers is required.

First Semester

Unit 1: Programming Basics	Programming languages. Development tools. Program design. Intro to P5.js
Unit 2: Graphics Programming	Drawing shapes and images. Keyboard and mouse control. Image processing. Sprite movement.
Unit 3: Physical computing	Arduino Platform. Controllable LED circuits, Breadboarding and prototyping. Timing control. Inputs and data acquisition
Unit 4: Integration project	Integration of graphics and physical computing in a single project.

Second Semester

Unit 5: Python	Syntax. Conditionals. Loops. Functions. Objects. Libraries and packages.
Unit 6: Console-based programming	Console and file IO. Input validation. Floating point calculations. Numerical algorithms. String processing.
Unit 7: Data structures	Lists. Tuples. Dictionaries. Heaps, stacks and queues. Trees. Graphs.
Unit 8: Object-oriented design	Encapsulation. Polymorphism. Inheritance. Top-down design. GUI programming.
Unit 9: Algorithm design and analysis	Linear and binary search. Naïve sort. Quick sort. Merge sort. Recursion.

PROGRAMMING FOR STEM/ENGINEERING

1. Materials
   Internet access, 1-to-1 computer use daily, and access to the LSU/BRBytes servers.

Hardware/Reusable Material	Recommended Unit	Cost/Unit
Arduino Starter Kits (Sunfounder Basic Starter Kit with Arduino Uno Board) https://www.sunfounder.com (Cannot be shared between sections)	1 per student	$50
Consumable Material		
Annual consumable materials to replenish Arduino supplies.	1/Classroom	$150
Software		
Arduino IDE	1 per student	Free on PC; $1/month for Chromebook

2. Required software, networking access, and access to LSU/BRBytes servers
   - Students will need to sign up with online development and testing environments, including but not limited to codesandbox.io, jsfiddle.net, scratch.mit.edu and others.
   - Students will need access to YouTube instructional videos relevant to the course, as well as other educational video repositories.
   - Teachers will need to be able to access the LSU/BRBytes servers using several Internet protocols including but not limited to HTTPS and SSH.
   - Students and teachers will access the curriculum and teaching materials through the LSU and BRBytes servers.
   - Teachers will need to share student data with their designated LSU Pathway Point-of-Contact.
   - Principals will need to communicate with the district's information technology department to ensure that there are no technological restrictions that block access to the LSU/BRBytes servers in the lsu.edu, brbytes.org or lsupathways.org domains on any port. In addition to the sites mentioned above, students will need web access to:

w3schools.com	codepen.io	tonejs.github.io
editor.p5js.org	create.arduino.cc	lunapic.com
elm-lang.org	youtube.com	github.com
freesound.org	p5js.org	stackoverflow.com

3. Required teacher collaborations
   Teachers will communicate with LSU instructors via emails and apps hosted on the LSU/BRBytes servers.
4. Required administration of course content, pre/post test, and research instruments
   All required materials and instruments will be either posted in the LSU/BRBytes servers or their location announced via email with the teacher/instructor group for this course.
5. Course Work
   Teachers must present the course material in sequence or as approved by collaboration with the LSU Pathway Point-of-Contact. The teacher is responsible for utilizing the LSU/BRBytes servers based system

to release, acknowledge, provide student feedback, and grade student work. The LSU/BRBytes servers will track and notify the teacher as students near the required 75% attainment mark for certification.

6. Other

   As this is a project-based learning class, we strongly suggest that each section of the course be limited to a *maximum* of 20 students. The course is dependent on the teacher providing feedback and reviewing student code. The course requires that teachers have adequate time to interact with each student.