Interactive Computing

Overview

This is a course for students in 10th grade and up. It focuses on the nuisances of programming for interacting with the real world in two representative areas: autonomous robots and the front end of web applications. Students learn how to iteratively approximate a software model to the realities of the physical hardware, how to write test suites and how to systematically debug their programs. Through fun and engaging projects, the students learn problem solving skills, such as programming robots to navigate mazes and play soccer, developing on-line pages to read sensors and control actuators in greenhouses, and automating devices at home with Internet of Things (IoT) technologies.

Objectives

- Explain how computers interact with humans.
- Create a simple interactive computer mini-app.
- Explain the fundamentals of electronic circuits.
- Demonstrate how to drive microprocessor-based circuits.
- Describe the basic syntax and semantics of Python.
- Describe the basic syntax and semantics of JavaScript.
- Demonstrate how to network a computer and a microprocessor.
- Combine programs running in different processors to cooperatively run an application.

Assessment

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

Course Essentials

Equipment	**Cost/Unit**
Classroom set of computers	$0 if you already have some, $500-600 per computer if you need to purchase
Reusable material for Arduino computing	$1,500
Microbit kits with necessary components (including wi-fi modules)	$55 per kit
Microbit-compatible robot car	$30-$60 per car
Prerequisite	Intro to Computational Thinking (LSU Partnership)

First Semester

Unit 1: Computer devices	Keyboard and mouse control. Image processing. Simple GUI programming.
Unit 2: Arduino	Arduino Platform. Controllable LED circuits, Breadboarding and prototyping. Timing control. Inputs and data acquisition Project: Control mini-greenhouse
Unit 3: Microbit and Python	Introduction to Python. Controlling Microbit boards with Python.
Unit 4: Microbit-based robot navigation	Navigation techniques. Maze solving strategies. Project: Navigation competition

Second Semester

Unit 5: Web technologies	Basic HTML, CSS, JavaScript and the Document Object Model
Unit 6: More JavaScript	Arrays and Objects. Packages. Running JS in a server. Node.js
Unit 7: Server-assisted computing	Interfacing microbits and/or arduinos to offload computation to a server. Fundamentals of computer communication: encodings, protocols and networking layers.
Unit 8: Microbit-based robot soccer	Hybrid microbit/server programming. IoT technologies. Controlling robots via the Internet. Project: Robot-soccer competition

INTERACTIVE COMPUTING

1. Materials

   Internet access, 1-to-1 device (chromebook acceptable), and access to LSU servers.

Hardware/Reusable Material	Recommended Unit	Cost/Unit
Raspberry Pi 3 Model B+ Starter Kit - 16 GB (CanaKit)	1 per 3-4 students	$99.95
Mouse/Keyboard/Monitor for each Raspberry Pi	1 per Raspberry Pi	Varies*
Wireless Access Point (TP-Link)	1 per classroom	$24.99
Any adapters for using existing school hardware (for monitors): HDMI/VGA adapter or DVI/HDMI adapter	1 per Raspberry Pi	$6-8
Micro Bit kits	1 per 1-2 students	$30
Micro Bit-compatible robot car	1 per 3-4 students	$30-70

   *Varies depending on the number of components needed ($20 for mouse/keyboard - $200 if monitor is also needed)

2. Required software, networking access, and access to LSU 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 servers using several Internet protocols including but not limited to HTTPS and SSH.
   - 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 servers in the lsu.edu, college-readiness.lsu.edu or stempathways.lsu.edu domains on any port. In addition to the sites mentioned above, students will need web access to other web sites, including, but not limited to, the following:

20q.net	aboutmyinfo.org	allthingsd.com	apcentral.collegeboard.org
Code.org	checkpagerank.net	Computer.howstuffworks.com	crypto.interactive-maths.com
Policy.hu	m.wikihow.com	www.colorcodepicker.com	Libguides.mit.edu
Repl.it	thingful.net	tonystrains.com	www.barefootcomputing.org
www.braingle.com	Pingtool.org	www.digitalattackmap.com	www.inventwithpython.com
bjc.edc.org	curriculum.csmatters.org	rapidtables.com	www.bbc.com
www.iplocation.net	www.dummies.com	www.googlefight.com	www.mediashift.org
www.ftc.gov	www.e-cartouche.ch	www.informationisbeautiful.net	www.nationalarchives.gov.uk
www.pbs.org	www.explainthatstuff.com	www.mathsisfun.com	www.nytimes.com
www.ted.com	www.geogebra.org	www-math.ucdenver.edu	www.us.norton.com
www.time.com	www.prchecker.info	www.sorting-algorithms.com	www.string-functions.com

www.youtube.com	www.w3schools.com	www.webgamesonline.com	www.worksheetworks.com
zdnet.com	www.zooniverse.com		

3. Required teacher collaborations
   Teachers will communicate with LSU instructors via emails, Google Drive, and/or apps hosted on the LSU servers. Teachers will need to share sample student work with their designated LSU Pathway Point-of-Contact.
4. Required administration of course content, pre/post test, and research instruments
   All required materials and instruments will be either posted in the LSU servers, Google Drive, or their location announced via email.
5. Course Work
   Teachers must present the course material in sequence or as approved by collaboration with the LSU Pathway Point-of-Contact. Teachers are expected to deliver a minimum of 80% of the course material.
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.