ECIS TECHNOLOGY CONFERENCE
Bavarian International School
20-22 MARCH 2015

www.bis-school.com/ecis-tech
PUTTING COOL TECHNOLOGY IN THE HANDS OF STUDENTS

Dan Reedy
LEY SIN AMERICAN SCHOOL
PUTTING COOL TECHNOLOGY IN THE HANDS and MINDS OF STUDENTS
WE STARTED WITH A "Simple" QUESTION
What would happen if you put cool technology in the hands of students...
...AND WAIT?
THEY LEARN!
WELL, SOME OF THEM DO
WHY DID WE CHOOSE SPECIFIC TECHNOLOGIES?
WHAT HAS GONE WELL?
WHAT COULD GO BETTER?
WHAT WILL WE CHANGE?
Hour of Code

Try the basics of computer science. Millions have given it a shot.

20 hour courses for
Elementary Students
ONE STUDENT'S INTEREST IN HOUR OF CODE WAS THE CATALYST.
LEARN.CODE.ORG
VISUAL PROGRAMMING

DRAG AND DROP CODE BLOCKS TO BUILD PROGRAMS
WEB BASED TECHNOLOGIES

▸ HTML
▸ CSS
▸ JavaScript
▸ various other technologies
Learn to code interactively, for free.
Learn to code interactively, for free.
Computer programming

Learn the fundamentals of programming with the popular JavaScript language and ProcessingJS library. Write your own programs and share them, explore programs made by others, and learn from each other’s programs!

Learn Programming

Create Program

Documentation

Browse Programs

Intro to JS: Drawing & Animation
In these tutorials, you'll learn how to use the JavaScript language and the ProcessingJS library to create fun drawings and animations. If you've never programmed before, start here to learn how!

Advanced JS: Games & Visualizations
Now that you know how to program in JavaScript and make basic drawings and animations, how could you use that knowledge to make games and visualizations?

Advanced JS: Natural Simulations
Learn how to use JavaScript, ProcessingJS, and mathematical concepts to simulate nature in your programs. These tutorials are a derivative of “The Nature of Code” book by Daniel Shiffman (natureofcode.com), used under CC BY-NC.

Meet the Professional
What can you do with computer science and programming skills once you've learned them? We've invited people from all around the world and the industry to introduce themselves to you. Find out how...
Computer Programming

Learn the fundamentals of programming with the popular JavaScript language and ProcessingJS library. Write your own programs and share them, explore programs made by others, and learn from each other's programs!

Learn Programming

Create Program

Documentation

Browse Programs

Intro to JS: Drawing & Animation
In these tutorials, you'll learn how to use the JavaScript language and the ProcessingJS library to create fun drawings and animations. If you've never programmed before, start here to learn how!

Advanced JS: Games & Visualizations
Now that you know how to program in JavaScript and make basic drawings and animations, how could you use that knowledge to make games and visualizations?

Advanced JS: Natural Simulations
Learn how to use JavaScript, ProcessingJS, and mathematical concepts to simulate nature in your programs. These tutorials are a derivative of "The Nature of Code" book by Daniel Shiffman (natureofcode.com), used under CC BY-NC.

Meet the Professional
What can you do with computer science and programming skills once you've learned them? We've invited people from all around the world and the industry to introduce themselves to you. Find out how...
RUBY WARRIOR

NEW LEVELS & EXPLOSIONS!!!

A TRIUMPHANT QUEST OF ADVENTURE, LOVE & DESTINY ALL WITHIN A FEW LINES OF CODE

VENTURE FORTH!
BLOC.IO/RUBY-WARRIOR/

A TRIUMPHANT QUEST OF ADVENTURE, LOVE & DESTINY
ALL WITHIN A FEW LINES OF CODE

VENTURE FORTH!
Learn HTML, CSS, iPhone apps & more

Learn how to build websites & apps, write code or start a business.
We ❤ Treehouse
WHAT CAN YOU DO WITH THIS STUFF?
Tangible & Exciting
Javascript was crucial
IN CONTROLLING ROBOTS
var Cylon = require('cylon');

Cylon.robot({
  connections: {
    sphero: { adaptor: 'sphero', port: '/dev/rfcomm0' }
  },

  devices: {
    sphero: { driver: 'sphero' }
  },

  work: function(my) {
    every((1).second(), function() {
      my.sphero.roll(60, Math.floor(Math.random() * 360));
    });
  }
}).start();

// Source: http://cylonjs.com/documentation/platforms/sphero
var Cylon = require('cylon');

Cylon.robot({
    connections: {
        ardrone: { adaptor: 'ardrone', port: '192.168.1.1' }
    },

    devices: {
        drone: { driver: 'ardrone' }
    },

    work: function(my) {
        my.drone.takeoff();
        after((10).seconds(), function() {
            my.drone.land();
        });
        after((15).seconds(), function() {
            my.drone.stop();
        });
    }
}).start();

// Source: http://cylonjs.com/documentation/platforms/ardrone/
var Cylon = require('cylon');

Cylon.robot({
  connections: {
    leapmotion: { adaptor: 'leapmotion' }
  },

  devices: {
    leapmotion: { driver: 'leapmotion' }
  },

  work: function(my) {
    my.leapmotion.on('hand', function(payload) {
      Logger.info(payload.toString());
    });
  }
}).start();

// Source: http://cylonjs.com/documentation/platforms/leapmotion
function onPoseEdge(pose, edge)
    if edge == "on" then
        if (pose == "waveIn" or pose == "waveOut") then
            post = conditionallySwapWave(pose)
            if pose == "waveIn" then
                previousSlide()
            end
            if pose == "waveOut" then
                nextSlide()
            end
        end
    end
    if pose == "doubleTap" then
        myo.lock()
    end
    if pose == "fingersSpread" then
        playSlideshow()
    end
end
ROBOTICS PLATFORMS
EASY TO PROGRAM
OTHER Cool TECHNOLOGY
define :left_hand do
  sync :tick
  with_fx :reverb do
    play_pattern [55, 55, 57, 59], amp: 0.6, divisor: 1
  end
end

define :harmonic do
  sync :tick
  sleep 8
  $root = ([52, 50, 48]-[$root]).sample
  $key = [:major, :minor].sample
  play $root, divisor: 1
end

define :root do
  sync :tick
  sleep 0.5
  play $root, amp: 0.5, divisor: 1
end

Sustain Time

In addition to specifying attack and release times, you may also specify a sustain time. This is the time for which the sound is maintained at full amplitude between the attack and release phases.

play 60, attack: 0.3, sustain: 1, release: 1
define :left_hand do
  sync :tick
  with_fx :reverb do
    play_pattern [55, 55, 57, 59], amp: 0.6, divisor: 1
  end
end

define :harmonic do
  sync :tick
  sleep 8
  $root = ([52, 50, 48]-[$root]).sample
  $key = [:major, :minor].sample
  play $root, divisor: 1
end

define :root do
  sync :tick
  sleep 0.5
  play $root, amp: 0.6, divisor: 1
end

1.1 Live Coding
1.2 Exploring the Interface
1.3 Learning through Play
2 Synths
  2.1 Your First Beeps
  2.2 Synth Parameters
  2.3 Switching Synths
2.4 Duration with Envelopes
3 Samples
  3.1 Triggering Samples
  3.2 Sample Parameters

Sustain Time
In addition to specifying attack and release times, you may also specify a sustain time. This is the time for which the sound is maintained at full amplitude between the attack and release phases.

play 60, attacks: 0.3, sustain: 1, release: 1
A quick comment about the hotspot
AFTER
WHAT HAS GONE WELL?
School Dashboard

25 Students

- 640 Badges Earned
- 649 Code Challenges Completed
- 1129 Quizzes Passed
- 41514 Points Earned

Active students per day: 7 of 25 active in the last 30 days.
Group Standings

Daniel Reedy
Earned 317 achievements

Tiffany Bonzon
Earned 317 achievements
IMMEDIATE FEEDBACK
INSPIRATION TO OTHERS
WHAT HAS GONE WELL?
WHAT COULD GO BETTER?
"What should I do?"
LACK OF FOCUS
SHARING

PROGRESS
WHAT WILL WE CHANGE?
INTRODUCE

CHALLENGES
HAND-PICKED EXERCISES
Thymio II - Let's Code (Advanced Beginner)

1. Open Studio for Thymio II. This is what you will see:
2. These buttons are used to load and execute the code which has been written
3. This window shows an overview of the state of the robot
4. Here you see the available functions and events which can be used in the code
5. Where we will write our code,
6. The constants can be created or looked at
7. Events can be added
8. This line tells us if our code is complete and correctly written
ENCOURAGE INVOLVEMENT
JOIN THE
COMMUNITY
Keep in Touch
invites you to:

Training with Joy Kesten


Why can’t education keep up with technology? How can we engage young women to join the STEM fields?

Friday April 17th 2015
Time: 9h-15h
Ticket: 100CHF* Location: Leysin American School
To purchase tickets please email: edresearch@las.ch
*price includes lunch and coffee/snack breaks

Sponsored by:
COME SKI
HTTP://SPEAKERDECK.COM/REEDY
DAN REEDY
DREEDY@LAS.CH
Thank You!