STANFORD UNIVERSITY EPGY Summer Institutes 2013
A program of Stanford Pre-Collegiate Studies

Introduction to Computational Thinking using Snap!

Instructor: Omoju Miller
omojumiller@gmail.com
Summer 2013.

CS + X for all X
Peter Norvig

Course Description: Students will be introduced to the fundamental thinking processes that underlie Computer Science through the use of a blocks programming interface called Snap!. In addition, we will discuss the Big Ideas of computing and a broader understanding of the role of computing in the 21st century. This course emphasizes the interdisciplinary nature of Computer Science and helps students prepare to be active participants in the broader computing field.
This course is broken up into two modules.
Module 1: We will be using the extended implementation of MIT's Scratch from University of California at Berkeley called Snap! (http://snap.berkeley.edu) to learn about computation thinking.
Module 2: We will be using robotics to enhance our understanding of the power of computing.

Tentative Course Outline:
The weekly coverage might change as it depends on the progress of the class. However, you must keep up with the reading assignments.

Day	Content
Day 1	• Intro to Blocks Programming and Big Ideas in Computer Science • Lab assignment: Student should create a GitHub Account • Lab assignment: Students should watch the BYOB tutorial to get acquainted with the environment at http://www.youtube.com/watch?v=Aub6BAxAT-c&feature=share&list=PLAE5AE3CD22628741 • Lab assignment: Students should play around with the Piano Project at the Scratch Website: scratch.mit.edu/projects/10012676/#editor
Day 2	• Random, If and Input • Introduction to Snap! aka BYOB • Project Introduction - Fractals • Lab assignment: Students should spend the time and decide on their proposal, create a presentation which they will pitch to the classmates next day • Lab assignment: Students should implement a version of a music app or extend someone else’s version
Day 3	• Algorithms: Solidify understanding of what an algorithm is and why theyre important to computer science and other fields. • Recursion: Cursory introduction • A bit about choosing and pitching a project • Lab assignment: Design an algorithm to solve project Euler problem 1, Multiples of 3 and 5 • Lab assignment:Type the pseudocode of you algorithm and push it to your GitHub account
Day 4	• Project Presentation • Recursion in Nature • The simplicity of complexity • Assembling Robots and helping TA’s out
Day 5	• Intro to Robotics • Robot Battle team formation and project • Lab Assignment: Make your robot touch the wall • The robot instructions for that can be found in file “Robot_touch_light_build_instruction.pdf”
Day 6	• Something interesting • Lab assignment: TBD
Day 7	• Something interesting • Lab assignment: TBD
Day 8	• Something interesting • Lab assignment: TBD
Day 9	• Robotics Project Presentation • Lab Assignment: Presentation on what I have learnt about computing