

Knight Foundation School of Computing and Information Sciences

Course Title: Computer Science Education for
Middle School Children

Date: 10/23/17

Course Number: IDC 4011C

Number of Credits: 4

Subject Area: Interdisciplinary
Computing

Subject Area Coordinator: Mark Weiss
email: weiss@cis.fiu.edu

Catalog Description: Provide teachers with the knowledge of **intermediate-level** Computer Science topics, as well as the pedagogy on how to teach the topics. Computer Science topics include computational thinking, logic, visual programming, computer hardware and networking, and social issues related to computer technologies including Internet safety.

Textbooks:

Online Curriculum: <https://studio.code.org/s/csd1> thru <https://studio.code.org/s/csd6> by Code.org

References:

<https://curriculum.code.org/csd/>

Prerequisite Courses: None

Corequisites Courses: None

Type: General Elective

Prerequisite Topics: (none)

Course Outcomes:

- O1. Be able to explain, create, follow, and debug algorithms to solve problems.
- O2. Demonstrate ability to develop web pages, build games, animations, and interactive art,
using a prototyping design process that identifies the needs of the user, designs a solution,
gets feedback, and repeats the cycle.
- O3. Be able to understand how data can be generated, analyzed, and used by society to solve
problems.
- O4. Research smart devices and explore sensors using Adafruit's Circuit Playground,
to learn
how to develop App Lab programs that utilize hardware inputs and outputs.

Knight Foundation School of Computing and Information Sciences
IDC 4011C

Computer Science Education for Middle School Children

This course should be taught by FIU faculty that have completed a 5-day Code.org workshop for Middle School Curriculum, as scheduled in the website:

<https://code.org/educate/professional-learning/cs-discoveries>

Outline

Topic	Number of Lecture Hours	Outcome
<ul style="list-style-type: none"> ● Problem Solving Process <ul style="list-style-type: none"> ○ Learn the Define-Prepare-Try-Reflect process <ul style="list-style-type: none"> ▪ Strategies to define open-ended problems ▪ Identifying I/O devices ▪ Develop and understand algorithms ▪ Learn the Develop-Test-Share model ▪ Learn the Input-Storage-Processing-Output model ▪ Propose and design an App to solve a real world problem 	5	O1
<ul style="list-style-type: none"> ● Software Programming <ul style="list-style-type: none"> ▪ Web Development <ul style="list-style-type: none"> ● HTML ● CSS ● Creating & Debugging Multi-page websites ● Creating a personal portfolio website ▪ Game Lab <ul style="list-style-type: none"> ● Drawing & Plotting Shapes ● Random Numbers ● Animating Sprites ● Loops & Conditionals ● Keyboard & Other Input to Games ● Collision Detection ● Functions ▪ Game Design Process 	20	O2
<ul style="list-style-type: none"> ● Data and Society <ul style="list-style-type: none"> ○ ASCII and Binary Representation ○ Encryption ○ Collection, Visualization of Data ○ Problem Solving, Interpreting, and Making Decisions with Data ○ Digital Footprints ○ Intellectual Property ○ Search Engines & Credible Sources 	10	O3
<ul style="list-style-type: none"> ● Hardware Programming <ul style="list-style-type: none"> ○ Using Circuit Playground, write programs 	10	O4

<ul style="list-style-type: none"> ▪ To Output: <ul style="list-style-type: none"> ● Blink an LED ● Make Noise with Buzzer ● Control a List of Color LEDs using For & Timed Loop ▪ Take Input: <ul style="list-style-type: none"> ● Buttons, switches ● Sound, Light, and Temperature Sensors ▪ To Prototype an Innovative Computing Device 		
---	--	--

Course Outcomes Emphasized in Laboratory Projects / Assignments

Projects and assignments will interactive lessons presented by students, as well as programming, projects done individually and collaboratively. Teaching demonstrations should be completed in a laboratory environment that includes short lectures by the instructor.

Outcome	
O1	Be able to explain, create, follow, and debug algorithms to solve problems.
O2	Demonstrate ability to develop web pages, build games, animations, and interactive art, using a prototyping design process that identifies the needs of the user, designs a solution, gets feedback, and repeats the cycle.
O3	Be able to understand how data can be generated, analyzed, and used by society to solve problems.
O4	Research smart devices and explore sensors using Adafruit’s Circuit Playground, to learn how to develop App Lab programs that utilize hardware inputs and outputs.

Oral and Written Communication:

- Written and oral discussions of social issues in computing

Theoretical Contents:

- Abstraction
- Algorithms

Problem Analysis Experiences:

None

Knight Foundation School of Computing and Information Sciences
IDC 4011C
Computer Science Education for Middle School Children

Solution Design Experiences:

- Weekly teaching labs, teaching lessons, programming/puzzles

Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan:
<https://abet.cs.fiu.edu/csassessment/>