DICE 3010 – Key Concepts in Computing

Course information
Credits: 5
Location/time: ADMN 203, W 6-8:05pm
In-class dates: 9/25, 10/9, 10/23, 11/6, 11/20, 12/4
Lab dates: 10/2, 10/16, 10/30, 11/13, 11/27

Instructor information
Instructor: Dr. Dylan Medina
Phone: (206) 296-5781
Email: medinady@seattleu.edu
Office: 910 12th Ave, LSAX 240

Course Description:
Learn the fundamentals of how computers and programming work by exploring the building-blocks of digital technology. From the logic of code to computational thinking, we’ll use simple “maker” projects and system design to learn through tinkering and exploration. We’ll also study the history of computing to discover the forces that have defined it both professionally and culturally. This is a hybrid course, with most course content delivered online along with face-to-face evening classes every two weeks.

Course Student Learning Outcomes:
Upon the successful completion of this course, students should be able to:

1. Explain foundational concepts of computational thinking such as logic, decomposition, pattern recognition, abstraction, algorithmic thinking, and evaluation.
2. Describe historical computing paradigms, including electrical, symbolic, textual, graphical, and embodied.
3. Assemble an Arduino microprocessor, articulate the roles of its input and output components, and program these functions using the Arduino C language.
4. Analyze and reverse-engineer a tangible computing project, documenting its basic system functions and logic.
5. Conceptualize and design a tangible computing system and model its primary use-case and system functions with a flowchart and 2D or 3D model.
6. Reflect on the impacts of computing history and consider your social values in relation to the potential for new computing paradigms.

E-Portfolios: students will produce a midterm and final project suitable for their program ePortfolio.

Instructional methods:
My approach to teaching is based on the "cognitive apprenticeship" model, which can be summed up by the sequence "I do [the teacher], we do [the class], you do [the individual student]." I will provide sample work and exemplars, we will discuss and work on your projects together during class times, and you will complete and present your work to the class.
Biweekly class sessions:
These sessions will be focused on: discussing course materials and collaborating on Arduino labs.

Biweekly check-ins:
These will be brief one-on-one video, phone, or email conferences for going over any questions or issues raised in the coursework, project planning, troubleshooting, and course feedback. These will be individually scheduled for a mutually convenient time.

REQUIRED READING:
All reading is available online or as worksheet/handout downloads from Canvas.

REQUIRED TECHNOLOGY:
2. Sunfounder Electronics Kit: https://www.amazon.com/gp/product/B00HEBHWWW/ (Links to an external site.)
A digital camera or smartphone for taking and uploading images to your Weebly course page.

REQUIRED SOFTWARE:
Arduino
The Sparkfun page below has a link to the Arduino download page and instructions for installing it on different operating systems:

Slack
You will receive an invitation to the DICE Slack Team Channel for this course. Please go through the quick tutorial if you are new to Slack:

COURSE SCHEDULE & TOPICS:

Week 1 (in class): What is Computational Thinking?

Topics explored in this week's presentation
☐ What is computing?
☐ Who gets to design computers?
☐ Computing and collaboration
☐ What is computational thinking?

What to read this week
To do before class
- Buy SparkFun Tinker Kit: https://www.sparkfun.com/products/13930
- Read the course syllabus and assigned reading
- Read/watch the presentation materials
- Note any questions you have so you can ask them in class
  - Join the course Slack team and If you’re new to Slack, please follow the tutorial prompts.

Week 2: Setting up the Arduino

Topics explored in this week's presentation
- Electronics and the Arduino
- Voltage, current, and resistance
- What is a microprocessor and what can it do?
- Why is a breadboard called a breadboard?

What to read this week

To do this week
- Do the reading
- Read/watch the presentation materials
- Download and install Arduino and connect to your Redboard

Week 3: (in class) Computer Models and System Models

Topics explored in this week's presentation
- What’s a computer model?
- What’s a system model and how do you flowchart one?
- Inputs and outputs
- What about the user?
- The Arduino programming cycle

What to read this week
Sparkfun Education "Voltage, Current, Resistance, & Ohm's Law". 
http://sparkfuneducation.com/concept-library/voltage-current-resistance.html

To do this week
☐ Do the reading
☐ Read/watch the presentation materials
☐ LAB: Circuit #1: Blinking an LED (Arduino syntax and LEDs)

Week 4: Programming basics and Arduino syntax

Topics explored in this week's presentation
☐ What is programming?
☐ Parts of a program
☐ Arduino and analog input/circuit bending

What to read this week

To do this week
☐ LAB: Circuit #2: Potentiometer (analog input)

Week 5 (in class): Control flow, input and output

Topics explored in this week's presentation
☐ Examples of control flow
☐ Arduino hardware
☐ Digital input and output
☐ Pulse-width-modulation

What to read this week

To do this week
☐ Do the reading
☐ Read/watch the presentation and tutorials
☐ LAB: Circuit #3: RGB LED (pulse width modulation)
ASSIGNMENT DEADLINE:
Midterm project: tangible computing reverse-engineering/analysis

Week 6: Decomposition

Topics explored in this week's presentation
☐ Cooking with decomposition
☐ Objects and encapsulation
☐ Decomposition in algorithms
☐ Arrays in Arduino

What to read this week
☐ ScienceBuddies, the engineering design process:
   http://www.sciencebuddies.org/engineering-design-process/engineering-design-process-steps.shtml#theengineeringdesignprocess

To do this week
☐ Read/watch the presentation
☐ LAB: Circuit #4: Multiple LEDs (arrays). Submit worksheets 1-4 to Canvas.
☐ Post peer feedback on midterms
☐ Post final project proposal

Week 7 (in class): Algorithms

Topics explored in this week's presentation
☐ Algorithms and socks
☐ Algorithm design and approximation
☐ Sorting
☐ Switches

To do this week
☐ LAB: Circuit #5: Push Buttons (switches)

Week 8 (optional lab): HCI and user-centered design

Topics explored in this week's presentation
☐ HCI and the non-expert user
☐ User-centered design
☐ From usability to sociability
☐ Mental models
To do this week
☐ Read/watch the presentation
☐ LAB: Circuit #6: Photo Resistor

Week 9 (in class): Networks and the Internet of Things

Topics explored in this week’s presentation
☐ What is the Internet of Things?
  • Who is the “smart city” for?
☐ Networks and data

To do this week
☐ Read/watch the presentation and tutorials
☐ LAB: Circuit #7: Temperature Sensor
☐ Present your final project storyboard (sketches on whiteboard or paper)

Week 10 Presenting projects

To do this week
☐ LAB: Circuit #8: A Single Servo

ASSIGNMENTS

15% Slack discussion and participation in class
20% Weekly Arduino Labs, documented in Weebly with an image and description
5% Midterm lab worksheet completion (1-4)
20% Midterm reverse engineering diagram and analysis
5% Final lab worksheet completion (5-8)
35% Final tangible computing system design with diagram, sketch and/or model, and description.

Slack discussion and participation in class
Class attendance is important and absences will affect your grade. For online weeks, we’ll stay in touch by posting and responding to Slack discussion questions.

Labs
Weekly Weebly-page posts describing and illustrating lab activities completed using the Arduino microprocessor (in class and at home) as described in the SparkFun tutorials. Your descriptions should state in your own words what the Arduino board and program
is doing. They should also state clearly any problems encountered with the lab and what steps you took to try and solve them.

**Deadline:** By class time (or during class if you have any problems) on weeks when we meet face-to-face. On non-class weeks, please complete the labs by Friday at 11:59pm.

**Lab worksheets:** these will help you understand the circuit and logic. Use them as needed and submit completed worksheets by midterm (#1-4) and at the end of term (#5-8).

---

**Midterm project**
Mid-term reverse engineering system flowchart and description (see description and rubric below).
Deadline: Week 5, April 27. Complete your work by class time so you can get feedback in class.

**Midterm circuit worksheets**
Complete and submit worksheets 1-4 to Canvas by Sunday April 30 at 11:59pm.

---

**Final Project**
A final project designs and model (see description and rubric below).
Deadline: Sunday June 11 by 11:59pm.

**Final circuit worksheets**
Complete and submit worksheets 5-8 to Canvas by Sunday June 11 at 11:59pm.

---

**Grading scale**

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100–94</td>
</tr>
<tr>
<td>B−</td>
<td>89–87</td>
</tr>
<tr>
<td>B</td>
<td>86–83</td>
</tr>
<tr>
<td>B+</td>
<td>89–87</td>
</tr>
<tr>
<td>C</td>
<td>76–73</td>
</tr>
<tr>
<td>C−</td>
<td>72–70</td>
</tr>
<tr>
<td>D</td>
<td>69–67</td>
</tr>
<tr>
<td>D+</td>
<td>69–67</td>
</tr>
<tr>
<td>C+</td>
<td>79–77</td>
</tr>
<tr>
<td>D−</td>
<td>62–60</td>
</tr>
<tr>
<td>F</td>
<td>59 or less</td>
</tr>
</tbody>
</table>

---

**Midterm project: Reverse engineering a tangible computing system**

The default project to describe for this assignment is Rob Faludi’s Botanicalls ([https://www.botanicalls.com/](https://www.botanicalls.com/)). You are free to find and propose a different tangible computing project to analyze and describe. Please do this **by week 3** by sending me an email with a link showing the project.

1. Begin your analysis by drawing a simple picture and writing an outline of what the system does. It should show the linear sequence of things that happen in order to trigger a phone call. Label each step in the process.
2. Once you have sequential description, think about the kind of information processing each step involves. What kind of information is input to the system? How is that information processed? What is the output? Make some notes about that data flow at each step in your sequence.

3. Now step back from your description and think about what it's modeling in the real world. When you keep a houseplant, what is the normal sequence of events involved in keeping it from dying? Write those down in a list.

4. Now you can think about how our computing concepts come into play. Write one or two sentences about how you think this system reflects each of these concepts:

   - Logic: how does the system use a logical sequence to keep the plant alive?
   - Decomposition: how is each step defined and dealt with?
   - Patterns: does the system look for patterns? This could be as simple as checking to see if sensor data is the same over a particular period of time.
   - Abstraction: what information is prioritized in the system? What's ignored?
   - Algorithmic thinking: what kind of logical sequence does the system repeat?
   - Evaluation: How might you check to see if this system was working well? What test-cases would you create to check the system with?

5. What is the mental model offered to users of this system? How do users interact with it? Why would someone use this system rather than sticking their finger in their plant's soil to try to figure out if it needs water? Do you think people talk to their plants? Why might they find it funny or gratifying to have their plant talk to them? Do you think this is a worthwhile system? Explain briefly why you think it is or isn't.

6. Finalize your sketch of the system, and turn your sequence of steps into an activity/system flowchart. Photograph and post these on your blog, along with your comments for each of the steps above. Each step should be no more than a couple of paragraphs or a paragraph plus a numbered list.

   Post your work on your blog, but save it as a DRAFT rather than publishing it. After I've assessed all the assignments I'll publish them for peer reviewing.

**MIDTERM PROJECT RUBRIC**

1. The system sketch and flowchart are clear and easy to understand. The sequence of steps described makes sense and captures the basic logic used

   Not really | Pretty | Yes!
   --- | --- | ---
   some key elements | much | All key elements
   are missing | are vague | and clear

   are some key elements | but some elements | are present
by the system. or missing

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2. The description of system input, output, and information processing make sense and capture all key elements.</td>
<td>Not really</td>
<td>Pretty much</td>
</tr>
<tr>
<td>3. The description of what real-world process the system models plausible and includes a list of steps in that process.</td>
<td>Not really</td>
<td>Pretty much</td>
</tr>
<tr>
<td>4. The sentences describing how this system reflects computing concepts are reasonable and show attention to concept definitions.</td>
<td>Not really</td>
<td>Pretty much</td>
</tr>
<tr>
<td>5. The mental model implied by how users interact with this system is plausible and an assessment of the system’s value is offered.</td>
<td>Not really</td>
<td>Pretty much</td>
</tr>
<tr>
<td>6. Your project sketch and flowchart are clearly photographed posted, along with your analysis, in draft form on your blog.</td>
<td>Not really</td>
<td>Pretty much</td>
</tr>
</tbody>
</table>

**Final project: Designing and modeling a tangible computing system**

Your written rationale for the project should include:

1. What the technology is for and who might use and benefit from it
2. How it relates to or re-imagines other computing technology
3. How you arrived at your idea by using the design engineering steps to:
   a. Define the Problem
   b. Do Background Research
   c. Specify Requirements
   d. Brainstorm Solutions
   e. Choose the Best Solution
4. A (very) preliminary sketch of how the system might function.
5. Be creative and don't be afraid to risk describing something cool even if you're not entirely sure about the details. If your idea is too ambitious, you can always scale it down later. No jetpacks.

The design should include:
1. a written description, activity & system flowchart, and 2D or 3D model of how the system works, including its inputs, outputs, system logic and at least one use case;
2. The activity and system flowchart should include separate swimlanes for user and system actions.
3. a description of the project’s user interface and interaction affordances, i.e how a user will know what it does and how to use it;
4. a description of how it demonstrates these key concepts of computing: logic, decomposition, abstraction, and algorithmic thinking);
5. a description of how input data is processed by the Arduino board to generate output. This should include some “pseudocode” approximating how the programming might work;
6. reflection on the learning and design process, and a description of the project’s limitations and possibilities for re-design
7. A response to peer feedback indicating how it contributes to your own reflections on learning and possible project revisions.

**FINAL PROJECT PROPOSAL RUBRIC**

<table>
<thead>
<tr>
<th>1. The proposal clearly states what this tangible computing technology is for and who might use and benefit from it.</th>
<th>Not really – some key elements are missing</th>
<th>Pretty much, but some elements are vague or missing</th>
<th>Yes! All key elements are present and clear</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. The proposal describes how this system design relates to and rethinks previous and current computing and/or technology designs.</td>
<td>Not really</td>
<td>Pretty much</td>
<td>Yes!</td>
</tr>
<tr>
<td>3. The proposal indicates how the project might be developed into a more complex and useful system.</td>
<td>Not really</td>
<td>Pretty much</td>
<td>Yes!</td>
</tr>
<tr>
<td>4. The proposal includes a simple, preliminary sketch of how the system might function.</td>
<td>Not really</td>
<td>Pretty much</td>
<td>Yes!</td>
</tr>
</tbody>
</table>

**FINAL PROJECT RUBRIC**

<table>
<thead>
<tr>
<th>1. The project includes a written description, activity &amp; system flowchart, and 2D or 3D model of how the system works,</th>
<th>Not really – some key elements are missing</th>
<th>Pretty much, but some elements are vague or missing</th>
<th>Yes! All key elements are present</th>
</tr>
</thead>
</table>
including its inputs, outputs, system logic and at least one use case.

| 2. The activity & system flowchart includes separate swimlanes for user actions and system actions. | Not really | Pretty much | Yes! |
| 3. The project includes a description of the project’s user interface and interaction affordances, i.e how a user will know what it does and how to use it. | Not really | Pretty much | Yes! |
| 4. The project includes a description of how it demonstrates these key concepts of computing: logic, decomposition, abstraction, and algorithmic thinking. | Not really | Pretty much | Yes! |
| 5. The design includes a description of how input data is processed by the Arduino board to generate output. This includes some “pseudocode” approximating how the programming might work. | Not really | Pretty much | Yes! |
| 6. The project includes reflection on the learning and design process, and a description of the project’s limitations and possibilities for re-design. | Not really | Pretty much | Yes! |
| 7. The project includes a response to peer feedback indicating how it contributes to your own reflections on learning and possible project revisions. | Not really | Pretty much | Yes! |

**Library and Learning Commons**
http://www.seattleu.edu/learningcommons/

**Academic integrity tutorial**
https://www.seattleu.edu/academicintegrity/
Support for students with disabilities
If you have, or think you may have, a disability (including an “invisible disability” such as a learning disability, a chronic health problem, or a mental health condition) that interferes with your performance as a student in this class, you are encouraged to arrange support services and/or accommodations through Disabilities Services staff located in Loyola 100, (206) 296-5740. Disability-based adjustments to course expectations can be arranged only through this process.

Missed classes
If you expect to be absent, please e-mail me well beforehand so I can set up a videoconference for you to attend synchronously. If for any reason you can’t be present or attend class via video, this will affect your grade unless you have a documented emergency.

Academic policies on the Registrar website
https://www.seattleu.edu/registrar/academics/performance/
Be sure that you understand the university policies below, posted on the Registrar’s website:
Academic integrity policy